

UNIVERSITÉ DE SHERBROOKE  
Faculté de génie  
Département de génie électrique et de génie informatique

# Architecture de contrôle d'un robot de téléprésence et d'assistance aux soins à domicile

Mémoire de maîtrise  
Spécialité : génie électrique

Sébastien LANIEL

Sherbrooke (Québec) Canada

Septembre 2019



# MEMBRES DU JURY

François Michaud

---

Directeur

Éric Beaudry

---

Évaluateur

François Ferland

---

Évaluateur





# RÉSUMÉ

La population vieillissante provoque une croissance des coûts pour les soins hospitaliers. Pour éviter que ces coûts deviennent trop importants, des robots de téléprésence et d'assistance aux soins et aux activités quotidiennes sont envisageables afin de maintenir l'autonomie des personnes âgées à leur domicile. Cependant, les robots actuels possèdent individuellement des fonctionnalités intéressantes, mais il serait bénéfique de pouvoir réunir leurs capacités. Une telle intégration est possible par l'utilisation d'une architecture décisionnelle permettant de jumeler des capacités de navigation, de suivi de la voix et d'acquisition d'informations afin d'assister l'opérateur à distance, voir même s'y substituer.

Pour ce projet, l'architecture de contrôle HBBA (*Hybrid Behavior-Based Architecture*) sert de pilier pour unifier les bibliothèques requises, RTAB-Map (*Real-Time Appearance-Based Mapping*) et ODAS (*Open embeddeD Audition System*), pour réaliser cette intégration. RTAB-Map est une bibliothèque permettant la localisation et la cartographie simultanée selon différentes configurations de capteurs tout en respectant les contraintes de traitement en ligne. ODAS est une bibliothèque permettant la localisation, le suivi et la séparation de sources sonores en milieux réels. Les objectifs sont d'évaluer ces capacités en environnement réel en déployant la plateforme robotique dans différents domiciles, et d'évaluer le potentiel d'une telle intégration en réalisant un scénario autonome d'assistance à la prise de mesure de signes vitaux.

La plateforme robotique Beam+ est utilisée pour réaliser cette intégration. La plateforme est bonifiée par l'ajout d'une caméra RGB-D, d'une matrice de huit microphones, d'un ordinateur et de batteries supplémentaires. L'implémentation résultante, nommée SAM, a été évaluée dans 10 domiciles pour caractériser la navigation et le suivi de conversation. Les résultats de la navigation suggèrent que les capacités de navigation fonctionnent selon certaines contraintes propres au positionnement des capteurs et des conditions environnementales, impliquant la nécessité d'intervention de l'opérateur pour compenser. La modalité de suivi de la voix fonctionne bien dans des environnements calmes, mais des améliorations sont requises en milieu bruyant. Incidemment, la réalisation d'un scénario d'assistance complètement autonome est fonction des performances de la combinaison de ces fonctionnalités, ce qui rend difficile d'envisager le retrait complet d'un opérateur dans la boucle de décision. L'intégration des modalités avec HBBA s'avère possible et concluante, et ouvre la porte à la réutilisabilité de l'implémentation sur d'autres plateformes robotiques qui pourraient venir compenser face aux lacunes observées sur la mise en œuvre avec la plateforme Beam+.

**Mots-clés :** Robot d'assistance mobile autonome, téléprésence, architecture décisionnelle



À ma conjointe et ma famille, Roxanne, Marie,  
Guy, Maxime et Cynthia



# REMERCIEMENTS

En premier lieu, j'aimerais remercier mon directeur de maîtrise, François Michaud, pour son soutien, son aide, sa compréhension et sa patience.

En second lieu, je désire remercier le réseau d'excellence pancanadien AGE-WELL pour le soutien financier apporté au projet.

Par la suite, j'aimerais remercier Dominic Létourneau pour son aide constante et ces idées innovatrices tout au long de ce projet de recherche. Merci également à Mathieu Labbé et François Grondin qui ont aidé à l'intégration et le soutien des bibliothèques RTAB-MAP et ODAS respectivement. J'aimerais aussi remercier tous les membres non cités de l'équipe d'IntRoLab qui se sont impliqués dans le projet.

Pour terminer, merci à ma mère, Marie Otis, qui a été une acolyte formidable lors des tests dans les domiciles.



# TABLE DES MATIÈRES

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Robots de téléprésence et d'assistance</b>	<b>5</b>
2.1	Fonctionnalités des robots de téléprésence . . . . .	5
2.2	Autonomie des robots . . . . .	8
2.3	Exemples de robots d'assistance aux soins et aux activités quotidiennes . . . .	9
2.4	Architecture décisionnelle pour robot autonome . . . . .	10
<b>3</b>	<b>Vers le déploiement d'un robot mobile de téléprésence pour l'assistance à domicile à distance</b>	<b>15</b>
3.1	Avant-propos . . . . .	15
3.2	Abstract . . . . .	17
3.3	Introduction . . . . .	17
3.4	SAM, our robot assistance platform . . . . .	19
3.5	RTAB-Map, ODAS and HBBA open source libraries . . . . .	21
3.5.1	RTAB-Map . . . . .	21
3.5.2	ODAS . . . . .	23
3.5.3	HBBA . . . . .	24
3.6	SAM's robot control architecture . . . . .	27
3.6.1	Sensors . . . . .	27
3.6.2	Autonomous navigation . . . . .	33
3.6.3	Autonomous conversation following . . . . .	38
3.7	Conclusion . . . . .	39
<b>4</b>	<b>Assistance autonome à la prise de mesure des signes vitaux</b>	<b>43</b>
<b>5</b>	<b>Conclusion</b>	<b>47</b>
<b>A</b>	<b>Caractéristiques des 10 domiciles</b>	<b>49</b>
<b>B</b>	<b>Analyse du comportement <i>Dock</i></b>	<b>57</b>
	<b>LISTE DES RÉFÉRENCES</b>	<b>59</b>





# LISTE DES FIGURES

1.1	Population estimée par groupe d'âge tous les 1er juillet . . . . .	1
2.1	Les robots de téléprésence décrits au tableau 2.1 . . . . .	7
2.2	Robot ACCOMPANY . . . . .	10
2.3	Architecture de contrôle du robot Donaxi . . . . .	12
2.4	Architecture décisionnelle du projet BonSAI . . . . .	13
2.5	Architecture décisionnelle du projet BonSAI . . . . .	13
2.6	Architecture HBBA ( <i>Hybrid Behavior-Based Architecture</i> ) . . . . .	14
3.1	SAM with its added components to the Beam+ platform . . . . .	22
3.2	Map generated by RTAB-Map . . . . .	23
3.3	ODAS architecture . . . . .	24
3.4	ODAS studio application . . . . .	25
3.5	Hybrid Behavior-Based Architecture (HBBA) . . . . .	26
3.6	SAM's robot control architecture using HBBA . . . . .	28
3.7	Teleoperation GUI . . . . .	29
3.8	2D representation of the path generated by the Dock behavior . . . . .	30
3.9	Face Following capabilities . . . . .	32
3.10	Navigation time histogram . . . . .	35
3.11	2D representation of the Environment A, B, E and J (left : Sketch, right : RTAB-Map) . . . . .	35
3.12	Kinect's blind spot on SAM . . . . .	37
3.13	Floor misinterpretation caused by vibration . . . . .	37
3.14	SAM's charging station . . . . .	38
4.1	Ajout de la motivation AVSM à l'architecture de contrôle de SAM . . . . .	44
A.1	Environnement A . . . . .	51
A.2	Environnement B . . . . .	51
A.3	Environnement C . . . . .	52
A.4	Environnement D . . . . .	52
A.5	Environnement E . . . . .	53
A.6	Environnement F . . . . .	53
A.7	Environnement G . . . . .	54
A.8	Environnement H . . . . .	54
A.9	Environnement I . . . . .	55
A.10	Environnement J . . . . .	55



# LISTE DES TABLEAUX

2.1	Capacités des robots de téléprésence . . . . .	6
3.1	Navigation results from 10 home environments . . . . .	33
3.2	Autonomous Navigation Through Door Frames . . . . .	37
3.3	Autonomous conversation following . . . . .	39
3.4	Description of the 10 Real Home Tested . . . . .	41
A.1	Description des domiciles où furent réalisés les tests . . . . .	50



# LISTE DES ACRONYMES

Liste des acronymes utilisés dans le présent document :

Acronyme	Définition
ADL	<i>Activities of Daily Living</i>
API	<i>Application Programming Interface</i>
AVSM	<i>Autonomous Vital Sign Monitoring</i>
BLE	<i>Bluetooth Low Energy</i>
CMC	<i>Crosswing Motion Control</i>
CSP	<i>Constraint Satisfaction Programming</i>
FOV	<i>Field Of View</i>
GUI	<i>General User Interface</i>
HBBA	<i>Hybrid Behavior-Based Architecture</i>
HRI	<i>Human Robot Interaction</i>
IMU	<i>Inertial Measurement Unit</i>
NUC	<i>Intel Skull Canyon NUC6i7KYK</i>
ODAS	<i>Open embeddeD Audition System</i>
RGB	<i>Red Green Blue</i>
RGB-D	<i>Red Green Blue and Depth</i>
ROS	<i>Robot Operating System</i>
RTAB-MAP	<i>Real-Time Appearance-Based Mapping</i>
SAM	Service d'assistance mobile
SPI	<i>Serial Peripheral Interface</i>
TRIC	<i>Telepresence Robot for Interpersonal Communication</i>
UdeS	Université de Sherbrooke
VSM	<i>Vital Sign Monitoring</i>



# CHAPITRE 1

## Introduction

Depuis plusieurs années, l'âge moyen de la population augmente dans la plupart des pays développés tels que le Canada. Comme le montre la figure 1.1, si la tendance se maintient, le nombre de personnes de 65 ans et plus dépasserait celui des enfants âgés de 15 ans et moins en 2019<sup>1</sup>. Avec ce changement démographique, certains besoins de la population seront plus en demande qu'auparavant. Les prestations de santé et l'aide à l'autonomie en sont des exemples. En effet, les personnes âgées souhaitent vivre à leur domicile le plus longtemps possible tout en étant autonome. Cependant, la perte de mobilité et la diminution des facultés provoquent une diminution ou même une perte de leur autonomie. De plus, étant plus fragile à cet âge, la nécessité d'être suivi par le personnel médical survient plus régulièrement. Les dépenses médicales ont énormément augmenté et 30% sont associées aux personnes âgées [1].

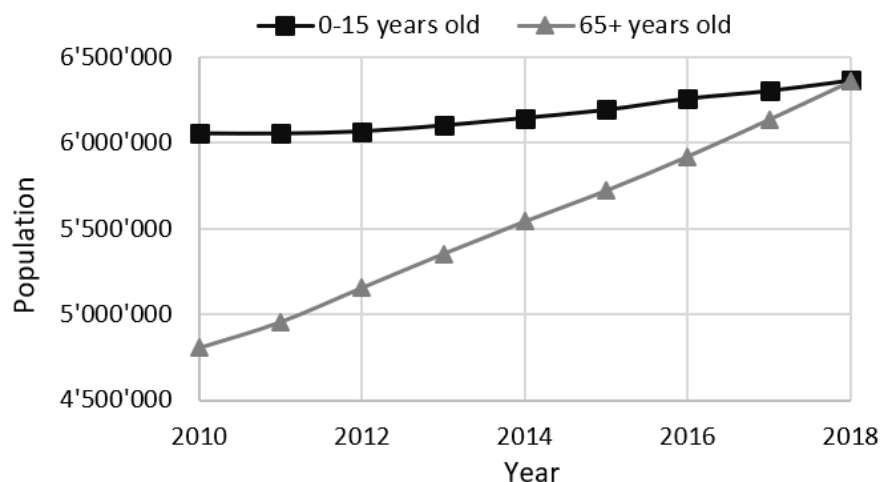


Figure 1.1 Population estimée par groupe d'âge tous les 1er juillet [2]

Afin de répondre à ces besoins, plusieurs recherches sont en cours quant à l'utilisation de la technologie dans les domiciles des personnes âgées. La domotique [3] a démontré des effets positifs permettant, par exemple, d'alerter l'équipe médicale après avoir détecté une chute à l'aide d'une caméra [4, 5]. Pour certains, la domotique deviendrait un standard dans les maisons et les préposés aux soins seraient des opérateurs à distance de robot d'assistance à domicile [6]. Cependant, cette alternative est présentement considérée trop coûteuse [7].

1. <https://doi.org/10.25318/1710000501-eng>

Les robots d'assistance aux soins et aux activités quotidiennes sont une alternative intéressante pour soutenir l'autonomie des personnes âgées. Une activité simple telle que la préparation d'un thé peut devenir une tâche sophistiquée pour une personne en perte de ses facultés cognitives. Un robot d'assistance peut aider cette dernière en lui expliquant les manipulations étape par étape [8]. Le robot doit toutefois faire preuve d'une certaine autonomie, dans des environnements qui peuvent présenter des situations complexes à gérer.

Parmi les autres solutions pour utiliser la technologie dans les domiciles, les robots de téléprésence ont également du potentiel en permettant au personnel de la santé et à la famille d'interagir à distance avec la personne âgée. En général, les robots de téléprésence, agissant comme des *Skype on wheels* [9], sont constitués d'un écran, d'haut-parleurs, d'une caméra, d'un microphone et d'une base motorisée. Le tout permet d'échanger les flux audio/vidéo et l'opérateur peut mouvoir le robot via une interface utilisateur. Le robot TRIC (*Telepresence Robot for Interpersonal Communication*) [10, 11] est un exemple d'une telle application. Dans les dernières années, une panoplie de robots de téléprésence a été développée [12]. Actuellement, dans un contexte d'entreprise, les robots de téléprésence répondent aux besoins des employés qui travaillent à distance [13, 14]. Ils peuvent visiter un collègue à son bureau ou participer aux réunions. Dans un contexte de soins à domicile, la téléprésence peut être utilisée par un médecin pour faire un suivi et peut permettre à un physiothérapeute de réaliser des séances de réadaptation. Cependant, les robots commerciaux actuels ne possèdent pas les éléments nécessaires pour accomplir une séance de téléadaptation. L'opérateur doit contrôler manuellement toutes les manoeuvres. Dans une situation de téléadaptation, cela peut devenir ardu pour le clinicien de réaliser simultanément sa séance et de contrôler manuellement un robot. À cet effet, le robot doit offrir les fonctions de base pour la téléprésence, transférer des données médicales en temps réel et offrir des fonctions autonomes pour diminuer la charge cognitive à l'opérateur.

Bien que les robots de téléprésence et les robots d'assistance aux activités quotidiennes sont des solutions envisageables pour répondre aux besoins d'aide à l'autonomie, les solutions actuelles démontrent des capacités validées individuellement. Il serait nécessaire de réunir ces capacités pour permettre une plus grande autonomie du robot. Dans ce mémoire, l'objectif principal est de réaliser une plateforme robotique qui unifie les éléments clés pour accomplir, de façon autonome, de la navigation et du suivi de conversation. La solution les offre à un opérateur dans un contexte de téléprésence et en contexte d'assistance autonome à domicile. Pour réaliser ces modalités, les bibliothèques en logiciel libre (*open source*) utilisées sont l'architecture comportementale HBBA *Hybrid Behavior-Based Architecture* [15, 16, 17] afin d'intégrer RTAB-Map *Real-Time Appearance-Based Mapping* (RTAB-MAP) [18, 19] pour la navigation et la cartogra-

---



---

phie, et ODAS *Open embeddeD Audition System* [20] pour la localisation, le suivi et la séparation de sources sonores. Le tout est caractérisé dans différents domiciles.

Ce mémoire est organisé comme suit. Le chapitre 2 présente la revue de littérature sur les fonctionnalités et les capacités des robots de téléprésence et d'assistance. Ce chapitre décrit également les modules nécessaires pour qu'un robot soit autonome et évalue la nécessité de concevoir une architecture de contrôle adaptée à l'intégration. Le chapitre 3 correspond à un article de journal, soumis à *Paladyn, Journal of Behavioral Robotics*, qui décrit SAM, la plateforme robotique développée. Ce chapitre explique l'intégration logicielle de l'architecture décisionnelle HBBA qui réunit les bibliothèques RTAB-Map et ODAS pour les modalités d'écoute auditive et de navigation respectivement. Cette intégration est évaluée dans différents domiciles réels et les résultats obtenus y sont dévoilés et expliqués. Le chapitre 4 présente un scénario autonome de prise de mesure des signes vitaux, poussant à la limite ce qui est envisageable de réaliser avec la plateforme SAM.

---



# CHAPITRE 2

## Robots de téléprésence et d'assistance

Ce chapitre recense les fonctionnalités des robots de téléprésence, identifie les éléments importants des robots autonomes, évalue les capacités des robots de soins à domicile et analyse la nécessité d'utiliser une architecture décisionnelle afin d'unifier les capacités requises pour l'autonomie décisionnelle d'un robot.

### 2.1 Fonctionnalités des robots de téléprésence

Au cours des dernières années, une multitude de robots de téléprésence ont été développés [12]. Ces derniers ont été conçus afin de répondre à des besoins commerciaux, médicaux ou à domicile. Ces différents robots possèdent plusieurs fonctionnalités similaires pour accomplir une séance de téléprésence. Le tableau 2.1 recense différents robots de téléprésence sur le marché ou en cours de développement, tels que le montre la figure 2.1. Tous les robots cités sont dotés d'une caméra permettant à l'opérateur de percevoir l'environnement, d'une base mobile contrôlée et, à l'exception du robot TRIC [10], sont équipés également d'un écran qui affiche le visage de l'opérateur. Les fonctionnalités recensées sont :

- **Base mobile omnidirectionnelle** : Certains robots tels que virtualME, RP-VITA et iRobot Ava, possèdent une base mobile omnidirectionnelle. Cela implique que le robot peut changer son orientation sans bouger son centre et, à l'inverse, se déplacer latéralement sans en changer son orientation. Cette capacité permet de reproduire la mobilité de l'humain. De plus, elle améliore la navigation du robot, car elle facilite les déplacements dans des endroits restreints ou entourés d'obstacles.
- **Stationnement autonome** : Cette fonctionnalité permet au robot de se garer à la station de recharge de façon autonome. L'utilisateur téléopère manuellement le robot pour le positionner à proximité de la borne et activer la commande. Cela permet au robot d'être plus convivial et améliore l'expérience [35].
- **Navigation autonome** : La navigation autonome permet à un robot de se déplacer par lui-même d'une position initiale à un endroit désiré sans que l'opérateur ait à intervenir. Cette capacité est intéressante car elle permet au robot d'accomplir des tâches sans l'intervention de l'opérateur. Par conséquent, sa charge cognitive due à la téléopération devrait se voir diminuer. Le côté négatif de cette fonctionnalité est l'obligation au robot d'avoir à

Tableau 2.1 Capacités des robots de téléprésence

Robots	Secteur	Omnidirectionnel	Stationnement autonome	Navigation autonome	Ajustement hauteur	Camera <i>pan-tilt</i>	Pointeur	Note	Références
Double 2	Commercial								[21]
Giraff	Commercial								[22]
Texai	Recherche								[23]
Beam	Commercial		✓						[24]
QB	Commercial				✓			Assiste l'utilisateur lors d'un passage au travers d'un cadrage de porte.	[25]
Vgo	Commercial					✓		La caméra est contrôlable et offre un zoom.	[26]
virtualME	Commercial	✓				✓		L'utilisateur peut pencher la caméra à distance.	[27]
Jazz Connect	Commercial		✓			✓		L'utilisateur peut spécifier une destination désirée visible à la caméra pour que le robot s'y déplace.	[28]
PRoP	Recherche					✓	✓	La caméra offre un zoom.	[29]
MantaroBot TeleMe	Commercial					✓	✓		[30]
MeBot	Recherche					✓	✓	L'utilisateur contrôle deux bras à trois degrés de libertés et le robot peut suivre le regard de façon autonome.	[31, 32]
RP-VITA	Commercial	✓	✓	✓				La caméra utilisée pour la navigation est indépendante de la caméra utilisée par l'utilisateur.	[33]
iRobot Ava	Commercial	✓	✓	✓	✓			La caméra utilisée pour la navigation est indépendante de la caméra utilisée par l'utilisateur.	[34]
TRIC	Recherche		✓	✓		✓	✓		[10]

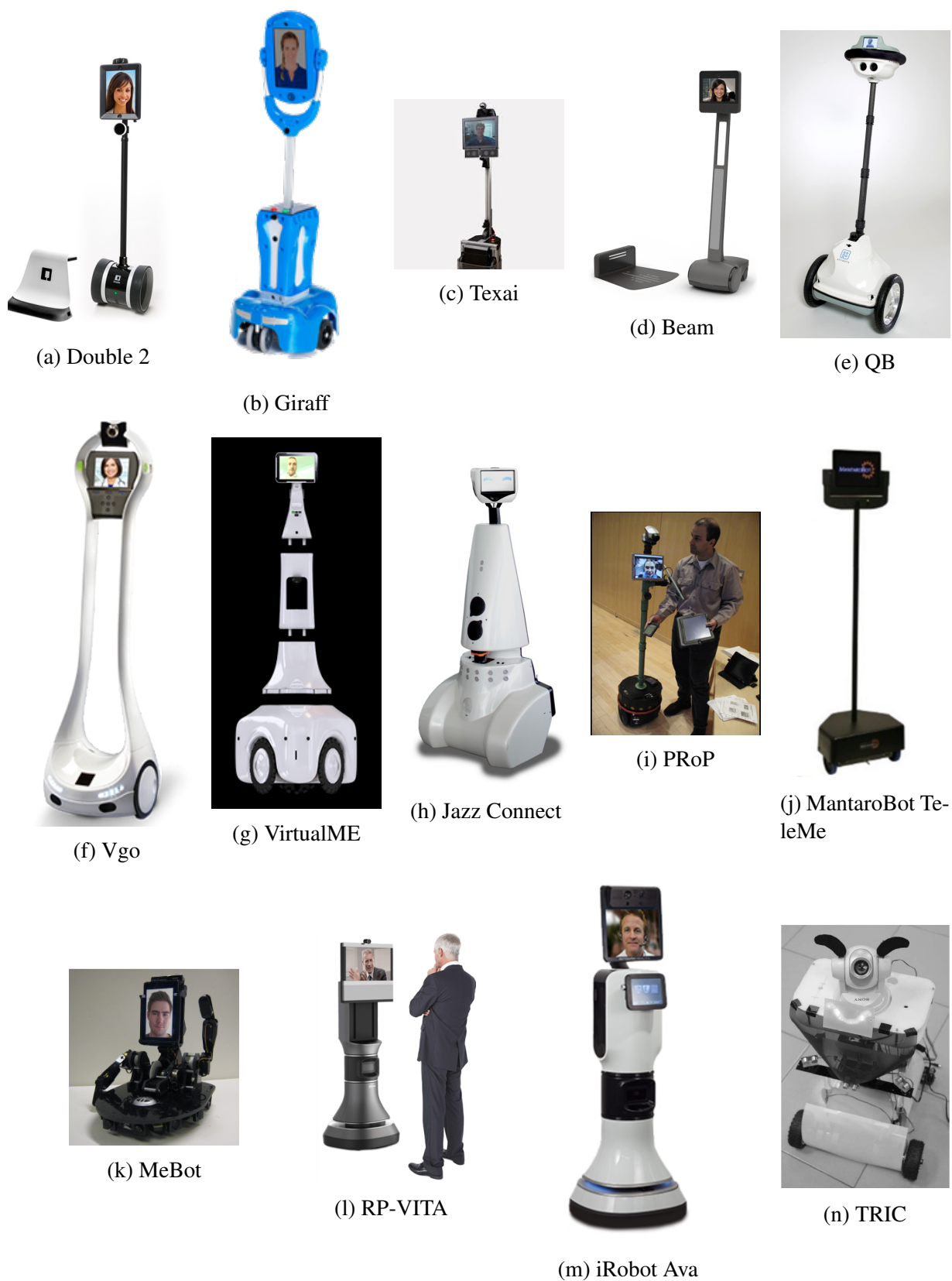


Figure 2.1 Les robots de téléprésence décrits au tableau 2.1

sa disposition davantage de modules de perception. Par exemple, pour réaliser un déplacement en étant autonome, le robot nécessite un ou plusieurs capteurs lui permettant de percevoir son environnement. Par la suite, cette information est analysée et le robot peut prendre des décisions en fonction des circonstances. Ainsi, s'il y a présence d'obstacles par exemple, le robot peut les éviter sans intervention de l'opérateur, ou encore pouvoir se rendre à un endroit spécifique.

- **Ajustement de la hauteur** : Cette capacité permet d'ajuster la hauteur de l'écran et de la caméra afin d'aligner l'écran au visage de son interlocuteur. Cette fonctionnalité donne l'impression d'être assis autour d'une table lorsque la hauteur est minimale et, à l'inverse, permet d'être en position debout lorsque nécessaire.
- **Caméra Pan-Tilt** : Ce dispositif permet à l'opérateur de contrôler la caméra dans les axes de tangage et de lacet. Cette capacité permet d'imiter la tête de l'être humain, car sans bouger le corps du robot, le téléopérateur peut déplacer le champ de vision de la caméra.
- **Pointeur** : À l'aide de cet outil, l'opérateur peut cibler un objet à l'aide d'un bras ou d'un pointeur laser pour lui permettre de diriger le regard de l'interlocuteur vers un objet ou un endroit. Cela favorise l'immersion de l'opérateur dans l'environnement, car il est possible pour lui d'expliquer des concepts en utilisant le langage non verbal en soutien. Le robot MeBot est un prototype intéressant, car l'opérateur utilise une manette permettant de recréer des mouvements sur les bras à trois degrés de liberté [32]. Ainsi, l'opérateur, en plus de pouvoir pointer, peut gesticuler pour alimenter son dialogue.

## 2.2 Autonomie des robots

Les robots présentés au tableau 2.1 possèdent des caractéristiques communes qui sont primordiales pour accomplir une séance de téléprésence. Cependant, la plupart de ces robots sont contrôlés par l'opérateur et offre très peu d'autonomie autre que des fonctionnalités assistées telle que le stationnement autonome ou l'aide au passage à travers d'un cadrage de porte. Par conséquent, c'est l'opérateur qui doit s'occuper de contrôler le robot et d'éviter les obstacles à l'aide de l'information fournie par une interface utilisateur. Afin d'utiliser un robot de téléprésence pour offrir un service de soin à domicile sans en affecter l'efficacité du personnel de la santé, il est primordial que le robot possède des fonctionnalités autonomes pour alléger la charge cognitive de l'opérateur.

À cette fin, les robots doivent avoir à leur disposition des modules de perception pour interpréter leur environnement d'opération. Cet ajout est essentiel pour améliorer le système d'analyse du robot et pour lui permettre d'être adaptatif. La navigation autonome est un exemple qui nécessite des modules de perception et d'analyse. Pour réaliser un déplacement sans l'intervention

d'un opérateur, le robot doit mémoriser des informations sur son environnement global et son environnement local. L'environnement global est décrit par une carte générée et qui est archivée dans la mémoire du robot. L'environnement local consiste aux éléments à proximité du robot. Il doit avoir des capteurs qui lui permettent de détecter les obstacles et de corriger son chemin à travers la carte globale. Plusieurs projets robotiques démontrent de telles capacités [36, 37, 38].

## 2.3 Exemples de robots d'assistance aux soins et aux activités quotidiennes

Le projet ACCOMPANY, développé par l'Université d'Hertfordshire, est un robot autonome qui assiste les personnes âgées à leur domicile [39]. Les objectifs de ce projet étaient de réaliser un robot qui est perçu comme un compagnon pour combattre l'isolement social et qui accompagne la personne pour accomplir des tâches quotidiennes afin de promouvoir l'autonomie. Le groupe de recherche a choisi et modifié le robot Care-O-Bot 3 illustré à la figure 2.2. Ce dernier possède une plateforme holonmique, un bras articulé de sept degrés de liberté, un bras de trois degrés de liberté qui contient une tablette tactile et plusieurs capteurs tels qu'une caméra et des capteurs laser. Une interface utilisateur indépendante permet à l'utilisateur de demander de l'assistance au besoin. Cette interface est reliée à une tablette électronique qui permet de faire une requête d'assistance à distance. Pour utiliser le plein potentiel du robot, la maison doit être adaptée en y ajoutant des caméras reliées au plafond, des commutateurs sur les portes et les armoires, des détecteurs de pression sur les sièges, etc. L'utilisation de caméra installée au plafond permet au robot de connaître en temps réel sa position et celle des autres personnes ou obstacles dans une pièce. Cette approche est intéressante, car elle fournit de l'information utile qui n'est pas toujours accessible par les capteurs situés sur le robot. Ainsi, à l'aide d'un algorithme de navigation, le robot peut se déplacer aux côtés d'une personne à une distance sécuritaire sans que les capteurs sur le robot la détectent. En observant les objets, les postures et les mouvements des utilisateurs, le robot reconnaît des situations à l'aide de modèles à l'interne [40]. Ainsi, la corrélation entre ses modèles et l'information de ses capteurs permet au robot d'inférer sur les actions des utilisateurs et de suggérer l'assistance appropriée. Par exemple, si le robot détecte qu'un invité a sonné à la porte, il propose à l'utilisateur de l'accompagner à l'entrée du domicile. Les détecteurs de pression sur les sièges permettent au robot de connaître la position de l'utilisateur. Ainsi, si l'utilisateur reste trop longtemps assis, le robot peut lui suggérer de faire une activité qui va le faire bouger. L'inconvénient d'utiliser des capteurs sur les meubles, les sièges et des caméras au plafond, c'est l'augmentation du prix au produit final. En effet, pour chaque client, il doit y avoir une installation et une adaptation du domicile. Actuellement, la majorité

des maisons ne sont pas munies de ces capteurs. Dans ces conditions, cette solution n'est pas idéale.



Figure 2.2 Robot ACCOMPANY

Le Giraff est un robot de téléprésence développé par Giraff Technologies AB. Son application primaire est d'offrir la téléprésence à domicile pour les personnes âgées [41]. Giraff est la plateforme robotique utilisée dans le projet GiraffPlus pour la téléprésence [42]. Ce projet consiste à domotiser une maison pour offrir de la sécurité, des suivis médicaux et de l'assistance aux activités quotidiennes des personnes âgées. Les capteurs de la maison, les capteurs médicaux et le robot de téléprésence sont reliés à un système nuagique nommé *GiraffPlus System*. Ce dernier récupère l'information prise par les différents capteurs et en génère une base de données. Le système permet de monitorer les activités du client, comme par exemple, évaluer la durée quotidienne qu'il a passé assis dans une chaise, détecter s'il a chuté ou détecter lorsqu'il change de pièce. Les capteurs médicaux nommés *Look4MyHealth* ont été développés par la compagnie IntelliCare. Cette solution permet au système d'archiver le poids, la tension artérielle, le taux de glycémie dans le sang, etc. Toutes ces informations sont mises à la disposition du personnel médical par l'entremise de la base de données. Les tests sur de vrais patients ont démontré que le système était bénéfique pour le suivi médical et pourrait être utilisé dans un milieu résidentiel. En 2013, le groupe de recherche prévoyait étendre leur installation dans 15 maisons.

## 2.4 Architecture décisionnelle pour robot autonome

Certains concours, tels que *RoboCup@Home* et *Rockin@Home*, demandent à des robots de service de réaliser des tâches d'assistance aux activités quotidiennes. Ainsi, pour que le robot



soit autonome et qu'il réalise ses objectifs, les développeurs unifient les capacités du robot dans une seule architecture décisionnelle.

*Team Homer@UniKoblenz* utilise une architecture composée d'une couche fonctionnelle, d'une couche comportementale et d'une couche d'application [43]. Cette architecture est implémentée dans deux robots semblables nommés *Lisa* et *The Lisa*. Ces robots démontrent, durant les compétitions de *RoboCup@Home*, des capacités telles que la détection et compréhension de dialogue, la détection et le suivi de personnes, la reconnaissance d'objets, etc. Cependant, l'architecture qui exécute les compétences semble répondre uniquement aux besoins du contexte de la compétition. En effet, pour des tâches complexes, les robots utilisent des machines à états finis pour suivre l'évolution des sous-tâches [44]. L'équipe de recherche souhaite dans leur travail futur améliorer la versatilité de leur architecture en y ajoutant de la gestion d'erreurs et de la rétroaction pour l'analyse de leurs sous-tâches. Cette solution peut être viable à la réalisation d'actions précises dans un milieu contrôlé tel que la compétition *Robocup@Home*. Cependant, cette approche est déficiente dans un réel domicile où l'environnement est dynamique et ne peut être parfaitement prédit. Pour terminer, il n'est pas spécifié si l'architecture est modulaire et/ou transportable sous différentes plateformes.

Le robot Donaxi a participé à Robocup@Home 2009, 2010 et 2012 [45]. Tout comme l'équipe *Homer@UniKoblenz*, l'objectif principal de ce robot est de répondre à des tâches précises à l'aide de différentes capacités. Comme illustré à la figure 2.3, l'architecture de contrôle est divisée en trois couches : délibérative, exécutive et fonctionnelle. Le niveau délibératif permet de représenter de façon symbolique et hiérarchique l'environnement. L'information recueillie permet la planification de tâches. La couche d'exécution et de contrôle est la couche centrale qui supervise l'exécution des plans initialement prévus. Cette couche permet de moduler les comportements du robot dépendamment de l'information reçue à partir de la couche fonctionnelle et des capteurs. Par exemple, le robot peut ralentir sa vitesse maximale si une situation dangereuse a été identifiée. Pour terminer, l'étage fonctionnel comprend plusieurs capacités nommées *Functional Group*. Chaque *Functional Group* permet la réalisation d'une tâche. Cependant, une tâche peut être réalisée à l'aide de différents *Functional Groups*. Par exemple, la navigation entre deux points peut se réaliser par un algorithme réactif, par un générateur de trajet ou par commandes manuelles fournies par l'utilisateur. Cette approche est intéressante car plusieurs méthodes peuvent être utilisées pour réaliser une même tâche et cela peut améliorer son adaptabilité. Par contre, il n'est pas spécifié comment la gestion de commande en conflit est réalisée. De plus, il n'est pas spécifié si l'architecture est transportable et modulaire pour faciliter son utilisation sur différentes plateformes et permettre l'ajout de nouvelles fonctionnalités avec facilité.

---

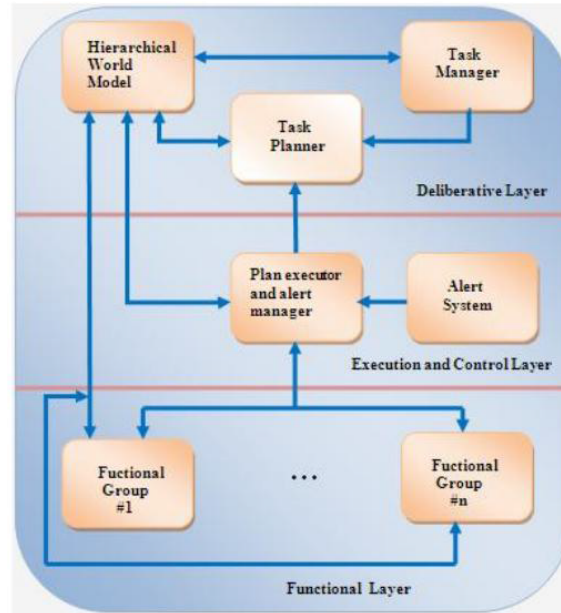


Figure 2.3 Architecture de contrôle du robot Donaxi [45]

Les concours tels que *RoboCup@Home* et *Rockin@Home* promouvoient le développement de robots pour exécuter des tâches de services à domicile et d'interactions humain-robot. Cependant, puisque l'évaluation réalisée lors du concours s'échelonne sur une courte durée, il est difficile de reproduire un contexte réel. Par conséquent, les architectures de contrôle développées ciblent davantage la réalisation de tâches précises pour gagner le concours plutôt que de cibler un développement générique et versatile pour répondre à un plus large besoin. Pour terminer, une problématique non traitée par les concours et par les équipes participantes est la gestion des ressources de calcul limitées des robots et la nécessité de moduler la demande en calcul des modules de traitement à l'intérieur même de l'architecture décisionnelle.

Pour faciliter la réutilisation d'une même architecture décisionnelle sous différentes plateformes robotiques, cette dernière doit être générique, modulaire et versatile. L'effort nécessaire pour adapter l'architecture d'un robot à l'autre doit être minimal. À cette fin, Siepmann et al. [46] proposent une architecture de contrôle qui isole la partie matérielle *Hardware Components* qui est propre au robot, comme le montre la figure 2.4. Cela facilite le transfert du code entre les plateformes. Les modules qui composent la couche matérielle sont ceux qui ont une connexion directe avec le robot et/ou qui ont besoin de fonctionner en temps réel. À noter que l'architecture est hiérarchique, ce qui veut dire que les modules peuvent échanger de l'information entre un module de même niveau ou inférieur. Ainsi, la couche fonctionnelle traite l'information provenant de la couche matérielle pour en extraire de l'information générique. Les composants du niveau fonctionnel sont également modulaires et facilitent l'ajout de nouvelles capacités. La couche BonSAI, illustrée à la figure 2.5, est l'étage haut niveau qui permet de moduler les com-

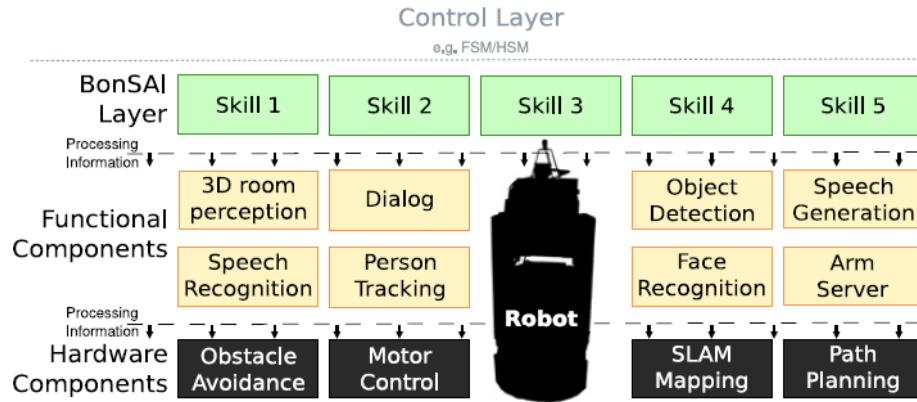


Figure 2.4 Architecture décisionnelle du projet BonSAI [46]

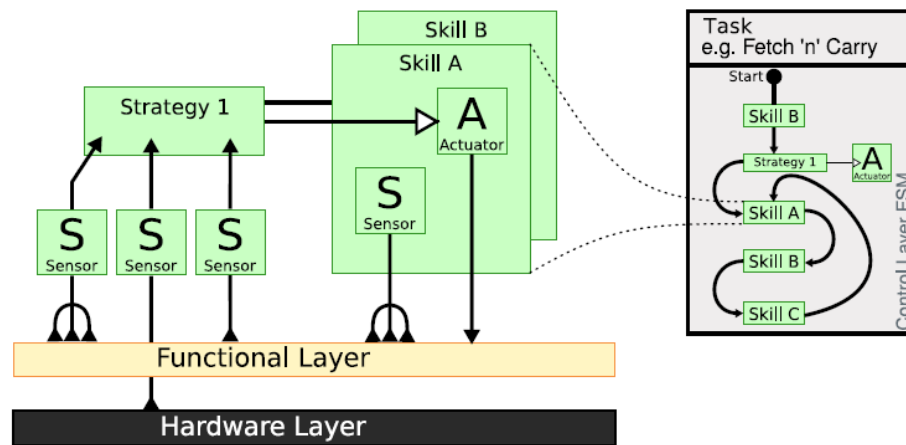


Figure 2.5 Architecture décisionnelle du projet BonSAI [46]

portements du robot à l'aide de stratégies et de compétences (*Skills*). Une compétence est un module comportemental qui est constitué d'interaction entre les capteurs et les actionneurs (*Actuators*). La couche sensorielle (*Sensor*) est définie par l'information générique qui provient de la couche fonctionnelle, et les actionneurs consistent en des commandes qui sont transmises aux couches inférieures. Ainsi, la prise de décision est indépendante des composantes du robot lui-même. L'information reçue des capteurs doit uniquement être traduite en information générique et l'inverse est aussi vraie pour les actionneurs du robot.

Une autre architecture décisionnelle qui répond au même besoin est l'architecture comportementale *Hybrid Behavior-Based Architecture* (HBBA) [16, 17]. Cette dernière propose une structure modulaire qui gère dynamiquement les modules de perception afin de limiter les traitements. Cette architecture comportementale permet la réalisation de tâches complexes telle qu'une visite touristique d'un établissement. La figure 2.6 démontre une instantiation de la structure d'HBBA. La partie matérielle qui est reliée au robot se résume aux couches d'entrées/sorties et sont iso-

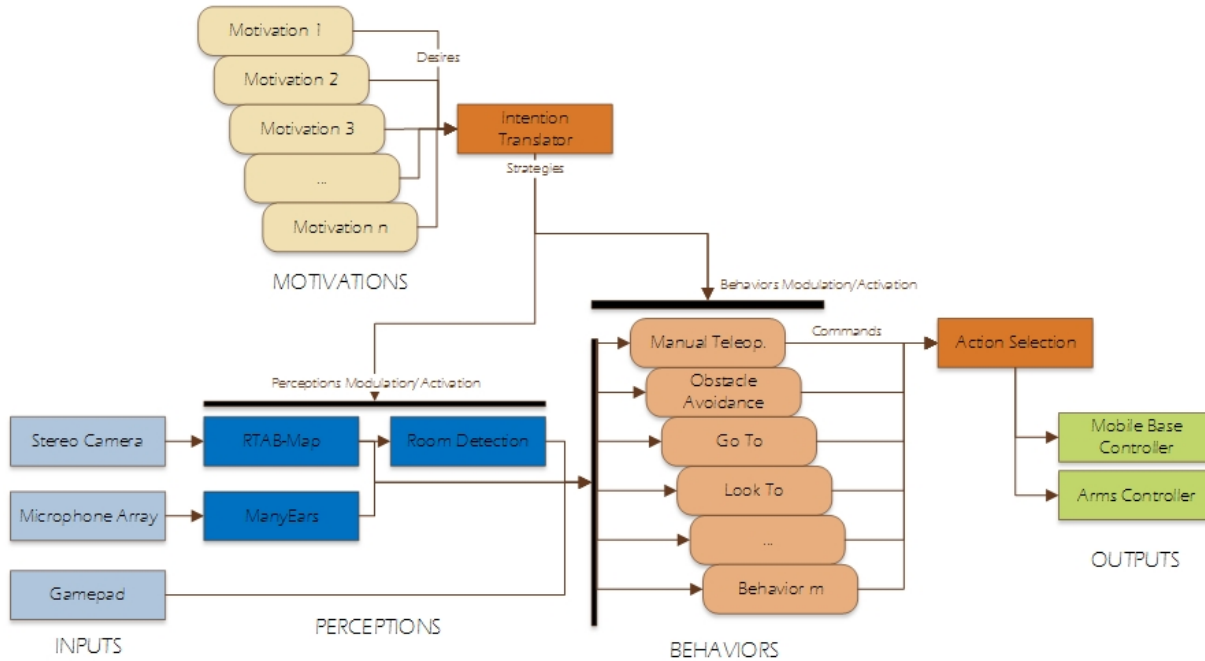


Figure 2.6 Architecture HBBA (*Hybrid Behavior-Based Architecture*)

lées de la structure décisionnelle. L'information des capteurs est analysée et est transformée en information générique par les modules perceptuels. Les comportements sont des fonctionnalités simples, tels que *Go To* et *Obstacle Avoidance*, qui permettent d'envoyer des commandes aux actionneurs. Les motivations modulent les comportements à l'aide de désirs qui sont transformés en intentions par le module *Intention Translator* selon un ensemble de stratégies spécifiques au robot utilisé.

HBBA et BonSAI sont des architectures décisionnelles qui possèdent du potentiel pour être transportables, modulaires et versatiles. L'utilisation d'une même architecture sur plusieurs plateformes robotiques qui accomplit différentes applications permettraient d'y valider son potentiel.

# CHAPITRE 3

## Vers le déploiement d'un robot mobile de téléprésence pour l'assistance à domicile à distance

### 3.1 Avant-propos

#### **Auteurs et affiliation :**

S. Laniel : étudiant à la maîtrise, Université de Sherbrooke, Faculté de génie, Département de génie électrique et de génie informatique.

D. Létourneau : professionnel de recherche, Université de Sherbrooke, Faculté de génie, Département de génie électrique et de génie informatique.

F. Grondin : stagiaire postdoctoral, *Massachusetts Institute of Technology, CSAIL Laboratory*.

M. Labbé : stagiaire postdoctoral, Université de Sherbrooke, Faculté de génie, Département de génie électrique et de génie informatique.

F. Ferland : professeur, Université de Sherbrooke, Faculté de génie, Département de génie électrique et de génie informatique.

F. Michaud : professeur, Université de Sherbrooke, Faculté de génie, Département de génie électrique et de génie informatique.

**Date de soumission :** 25 juillet 2019

**Revue :** *Paladyn, Journal of Behavioral Robotics*

**Titre anglais :** Towards the Use of a Telepresence Mobile Robot for Remote Home Care Assistance

**Titre français :** Vers le déploiement d'un robot mobile de téléprésence pour l'assistance à domicile à distance

**Contribution au document :**

Le chapitre 3 présente en détails la mise en œuvre du robot de téléprésence et d'assistance aux soins SAM, réalisé dans le cadre de ce mémoire. Ces travaux impliquent l'adaptation matérielle/logicielle de la plateforme robotique Beam+, l'utilisation des bibliothèques HBBA, RTAB-Map et ODAS pour l'implémentation des capacités requises par le robot, et la validation de cette implémentation dans des domiciles.

### **Résumé français :**

L'une des plus grandes transformations démographiques et sociales à laquelle sont confrontés les pays développés est le vieillissement de la population, l'augmentation de l'espérance de vie et les défis qui en découlent. Cela soutient le développement rapide des technologies utilisées pour fournir des soins à domicile, incluant la robotique. Dans le secteur de la santé, un robot de téléprésence pourrait être utilisé par un clinicien ou un aide soignant pour aider les personnes âgées à domicile sans être physiquement au même endroit. Cependant, la convivialité de ces plateformes pour de telles applications nécessite qu'elles puissent gérer la navigation et les interactions avec une certaine autonomie. Cela nécessite l'intégration de capacités décisionnelles autonomes sur une plateforme robotique équipée de capteurs et d'actionneurs appropriés, le tout validé dans des domiciles. Cet article présente l'intégration de trois bibliothèques libre d'accès sur une plateforme de téléprésence Beam+. Ces bibliothèques offrent la navigation autonome, l'audition artificielle et l'architecture de contrôle de robot. Elles ont été développées en tenant compte des exigences en temps réel, de puissance de calcul limitée et de robustesse, afin de pouvoir travailler en laboratoire et dans différents foyers. La validation de la plateforme résultante, baptisée SAM, est présentée basé sur des essais menés dans 10 domiciles.

## 3.2 Abstract

One of the greatest demographic and social transformations facing developed countries is caused by the aging of their populations, increased life expectancy and related challenges. Incidentally, this supports the fast growing development of technology used to provide home care, including robotics. In health care, a telepresence robot could be used to have a clinician or a caregiver assist seniors in their homes without having to travel to these locations. However, usability of these platforms for such applications requires that they can navigate and interact with some level of autonomy. This requires the integration of autonomous decision-making capabilities on a platform equipped with the appropriate sensing and action modalities, validated in real home settings. This paper presents such integration on a Beam+ telepresence platform and using three open source libraries for autonomous navigation, artificial audition and integrated robot control architecture, developed with real-time, limited processing and robustness requirements in mind so that they can work out of the lab and into real homes. Validation of the resulting platform, named SAM, is presented from trials conducted in 10 homes.

## 3.3 Introduction

All around the world, problems caused by population aging drive interest in developing technology, including robotics [47, 48], to provide home care. In particular, mobile telepresence robotic platforms are now commercially available and provide mobility to sensors, effectors and interactive devices for usage in real world settings, without having to modify the environment. These platforms usually consist of a mobile base, a camera, a screen, loudspeakers and a microphone, making them mobile videoconference systems, commonly referred by some to be “Skype on wheels” [9]. Telepresence robot’s success in telehealth applications in hospitals and homes [49] depend on their ability to maximize the quality of care provided remotely, i.e., providing diagnostics and treatments of the same quality compared to face-to-face or conventional means, and to maximize the benefits for the patients, the clinicians and the system of care. Home telehealth, or telehomecare, consists of providing health care services into a patient’s home [50]. Services can be very diversified and include telecheckups (e.g., phone calls to ensure that the patient is doing well), telemonitoring by recording physiological data, teleconsultation, and teletreatment [51, 52, 53]. Teleconsultation and teletreatment, also referred to as tele-visits or video visits, require more lively and interactive virtual encounters using both video and audio, synchronized for remote, live consultations between patients, clinicians and caregivers [50]. Video visits are intended to become an alternative to traditional face-to-face assessments or treatments : when doing so, patient satisfaction is important in maintaining motivation and treatment compliance,

---

and the satisfaction of healthcare professionals must be high for new treatments to become mainstream in clinics [54].

Consequently, telepresence mobile robots [12][55] can certainly provide interesting benefits for telehomecare visits, vital sign monitoring and Activity of Daily Living (ADL) assistance [10], giving mobility to telecommunication and monitoring devices. These benefits can only be possible if the remote users can do what they have to, using the sensing and acting capabilities of the robot platform. To do so, two key elements that must be taken into consideration are minimizing cognitive load and maximizing situation awareness [56], by limiting what the remote operators have to do to control the platform while still perceive the appropriate information to focus on the interaction tasks to be conducted through telepresence. And evidently, all of these services must be provided at minimal cost.

We decided to take on this challenge by targeting four principal requirements needed to assist the remote operator, whom would most likely be novice robot users (e.g., clinicians, caregivers) conducting such virtual visits, to avoid having the occupant provide assistance to the robot, and to be able to adapt to technological evolution and availability of platforms on the market :

- **Autonomous recharging.** The robot must be able to recharge autonomously by navigating to its charging station. This is an essential feature to avoid having the remote operator teleoperate the robot back to its charging station at the end of a session, or to be moved by the occupant in case of a telecommunication failure or low energy level. This requires the robot to have mapping and localization capabilities, allowing it to navigate efficiently and safely in the home.
  - **Efficient interaction with people.** To minimize cognitive load and maximize situation awareness [56], we believe that a remote operator would find it beneficial to receive assistance in following and tracking people which whom to interact. Such capabilities would minimize what the remote operators have to do to control the platform and focus on the interaction tasks to be conducted through telepresence.
  - **Expandability and portability.** With continuous technological progress and the availability of new mobile robot platforms having higher processing and interacting capabilities, it is important to use a robot control integration framework (a.k.a. architecture) allowing to expand and port the system in accordance with user's needs and the sensing, reasoning and acting capabilities of the robot platform.
  - **Testing in real home settings.** Currently, related work conducted in real home environments do not underline the difficulties encountered and the limitations of their designs [47, 48] or worse, are not evaluated in real environments at all [7, 57, 58, 59]. To move
-



toward the use of telepresence mobile robots for remote home care assistance, it is therefore important to conduct trials in such conditions. Doing so before conducting trials with typical users makes it possible to outline what can be expected of the robot, and derive interaction scenarios in accordance with the robot's capabilities.

This paper presents how we satisfy these requirements using three open source libraries : RTAB-Map [18, 19] for navigation, ODAS [60] for sound processing, and HBBA [61, 17] as the robot control architecture. These libraries are developed by always taking into consideration the constraints of having to operate in real world settings using robots that have limited processing capabilities. We illustrate their use in the design of SAM [62], our remote assistance robot platform designed on a Beam+ platform from Suitable Technologies Inc. Results from trials conducted in 10 home environments are presented, identifying recommendations on what to improve or to take into consideration before conducting usability studies with SAM.

The paper is organized as follows. First, Section 3.4 presents SAM and its capabilities, followed by Section 3.5 with the open source libraries used. Then, Section 3.6 explains the implementation of SAM's robot control architecture using the open source libraries, the experimental methodology and observations made. Section 3.7 concludes the paper and outlines perspectives for future work.

## 3.4 SAM, our robot assistance platform

In recent years, a multitude of telepresence robots have been developed [12][55] to meet commercial, health or home care needs. For home care, telepresence robots should be lightweight (to facilitate their installation and their manipulation), stable (to avoid potential hazard in case of hardware failure or physical contacts) and inexpensive. Semi-autonomous navigation (e.g., traversing a doorway, avoiding obstacles, point-and-click path planning) or autonomous navigation (e.g., moving to a specific location outside the field of view of the robot) are available on robots usually equipped with laser range finders, which increase the price of the platform. A cheaper solution is to use a RGB-D camera sensor, such as the Microsoft Kinect for Xbox One camera. In addition, most platforms use differential drive locomotion with some being self-balanced using only two wheels, making them unstable if someone tries to lean onto it. Omnidirectional locomotion facilitates navigation in tight spaces, which could be quite useful in homes if the cost of the platform remains low. UBBO Maker robot has such capability, but has limited payload to add sensors for autonomous navigation or vital sign monitoring. Only the GiraffPlus initiative focuses on combining mobile virtual visits with remote vital sign monitoring, by having designed their own middleware for interfacing sensors [63, 64, 41]. The Giraff robot is used to

---

provide security, medical follow-up and assistance to daily activities of seniors. Home sensors, medical sensors and the Giraff robot are connected to a cloud-based system to retrieve the information taken by the various sensors to monitor the patient's activities, e.g., evaluates the daily time spent sitting on a chair, detects in which room the elder is, monitors weight, blood pressure and blood glucose levels. Despite the robot's limited capabilities, trials with real patients suggest that the system can be beneficial for medical follow-up and could be used in a residential setting.

After having reviewed the different telepresence platforms [62] and to avoid having to design a new one from the ground up, we chose to add sensing and processing capabilities on the Beam+ platform from Suitable Technologies. When we started the project, it was one of the least expensive platform (US\$2,000). It could be interfaced with the library presented in [65] to control the motors with velocity commands and to read odometry. The Beam+ platform comes with a 10" LCD screen, low power embedded computer, two  $640 \times 480$  HDR (High Dynamic Range) wide angle cameras facing bottom and front, loudspeakers, four high quality microphones, WiFi network adapter, a 20 Ah sealed lead-acid 12 V battery capable of approximately two hours of autonomy. The Beam+ robot also comes with a charging station : the operator just has to position the robot in front of it and activate the docking mode to let the the robot turn and back up on the charging station. The robot's dimensions are 134.4 cm (H)  $\times$  31.3 cm (L)  $\times$  41.7 cm (D). The Beam+ also comes with wheel encoders and an inertial measurement unit to estimate the change of position of the robot over time. Motor control and power management are accomplished via an USB 2.0 controller in the robot's base, and its maximum speed is 0.45 m/s.

To address the requirements outlined in Section 3.3, we needed to add sensors and processing capabilities to the Beam+ platform. As shown by Fig. 3.1, we placed a Kinect camera on top of the LCD screen using custom made aluminium brackets. The Kinect is placed at the top of the robot facing forward and slightly inclined to the ground to prevent SAM from hitting hanging objects or shelves. We installed a circular microphone array using a 8SoundsUSB [60] sound card and customized aluminum brackets and acrylic support plates at 67 cm from the ground. We added an Intel Skull Canyon NUC6i7KYK (NUC) computer equipped with a 512 GB hard drive, 32 GB RAM, a quad Core-i7 processor, USB3 ports, Ethernet and WiFi networking. We replaced the head computer's hard drive with a 128 GB mSATA drive. Both computers run the Ubuntu 16.04 operating system with ROS (Robot Operating Systems [66]) Kinetic. A low-cost USB dongle is installed on the robot to acquire the following vital signs from battery-powered Bluetooth Low Energy (BLE) sensors : blood pressure, SPO2 and heart rate, temperature, weight scale and glucometer [67]. We electrically separated the added components and the original robot by using a SWX HyperCore 98Wh V-Mount lithium-ion battery placed on the robot's base using a V-Mount battery plate, keeping the robot's center of gravity as low as possible

---

and facilitating battery swapping for charging. This allows us to revert any changes and to keep our modifications as less intrusive as possible. Coupled with DC-DC converters, the battery provides power to the microphone array, the Kinect and the NUC computer. The lithium-ion battery is recharged manually and separately. This configuration gives 50 minutes of autonomy when the robot maps its environment and 75 minutes in localization mode when using navigation modalities (i.e., autonomous navigation, teleoperation). Overall, the additional components plus the initial robot platform cost a total of US\$4,300.

## 3.5 RTAB-Map, ODAS and HBBA open source libraries

Autonomous navigation, artificial audition and robot control framework are research endeavours on their own. To benefit from progress made in these areas and to be able to focus on the integration challenge in designing a robot for remote home care assistance, we use open source libraries designed by our research laboratory and used in the research community. These libraries are designed with online processing and real world constraints in mind, and by being open source they provide replicability of the experimental methodology.

### 3.5.1 RTAB-Map

SPLAM (Simultaneous Planning, Localization And Mapping) [68] is the ability to simultaneously map an environment, localize itself in it and plan paths using this information. This task can be particularly complex when done online by a robot with limited computing resources. A key feature in SPLAM is detecting previously visited areas to reduce map errors, a process known as loop closure detection. For usage in home settings, the robot must be able to deal with the so-called kidnapped robot problem and the initial state problem : when it is turned on, a robot does not know its relative position to a map previously created, and it has, on startup, to initialize a new map with its own referential ; when a previously visited location is encountered, the transformation between the two maps can be computed. Appearance-based loop closure detection approaches exploit the distinctiveness of images by comparing previous images with the current one. When loop closures are found between the maps, a global graph can be created by combining the maps into one. However, for large-scale and long-term operation, the bigger the map is, the higher the computing power required is to process the data online if all the images gathered are examined. With limited computing resources on mobile robots, online map updating is limited, and so some parts of the map must be somewhat forgotten.

---

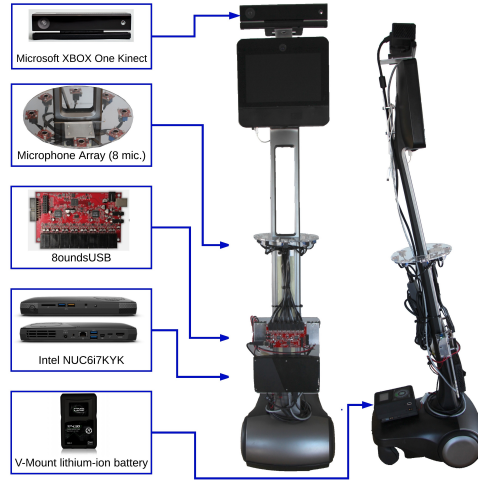


Figure 3.1 SAM with its added components to the Beam+ platform

Memory management approaches can be used to limit the size of the map so that loop closure detection is always processed under a fixed time limit, thus satisfying online requirements for long-term and large-scale environment mapping. RTAB-Map (Real-Time Appearance-Based Mapping)<sup>1</sup> [18, 19] is our open source library implementing such an approach, using images of the operating environment. Released in 2013, RTAB-Map can be used as a cross-platform standalone C++ library and with its ROS package<sup>2</sup> to do 2D or 3D SLAM. The standalone binaries have been downloaded more than 13,000 times and its GitHub repository has around 20 clones and 600 visits per day. RTAB-Map has been also used by the winners of the Microsoft Kinect Challenge held at IROS 2014<sup>3</sup>.

Figure 3.2 illustrates an example of a 3D representation and a 2D map representation created with RTAB-Map using a Kinect camera and a 2D lidar. The Kinect camera generates a depth image coupled with a standard RGB image, resulting in a colored 3D point cloud. The RGB image is also used to calculate image features stored in a database. RTAB-Map combines multiple point clouds together with transforms (3D rotations and translations) from one point cloud to the next. Estimation of the transforms are calculated from the robot's odometry using wheel encoders, visual odometry or sensor fusion [69]. Image features from the current image are compared to the previously calculated image features in the database. When the features have a strong correlation, a loop closure is detected. Accumulated errors in the map can then be minimized using the new constraint leading to a corrected map [70]. As the map increases in size, loop closure detection and graph optimization take more and more processing time. But RTAB-Map's memory management approach transfers, when a fixed real-time limit is reached, oldest and less

1. <http://introlab.github.io/rtabmap>

2. [http://wiki.ros.org/rtabmap\\_ros](http://wiki.ros.org/rtabmap_ros)

3. <http://goo.gl/UJiPzZ>

seen locations into a long-term memory where they are not used for loop closure detection and graph optimization, thus bounding the map update time to a determined threshold. When a loop closure is found with an old location still in working memory, its neighbor locations are brought back from the long-term memory to the working memory for additional loop closure detection and to extend the current local map.

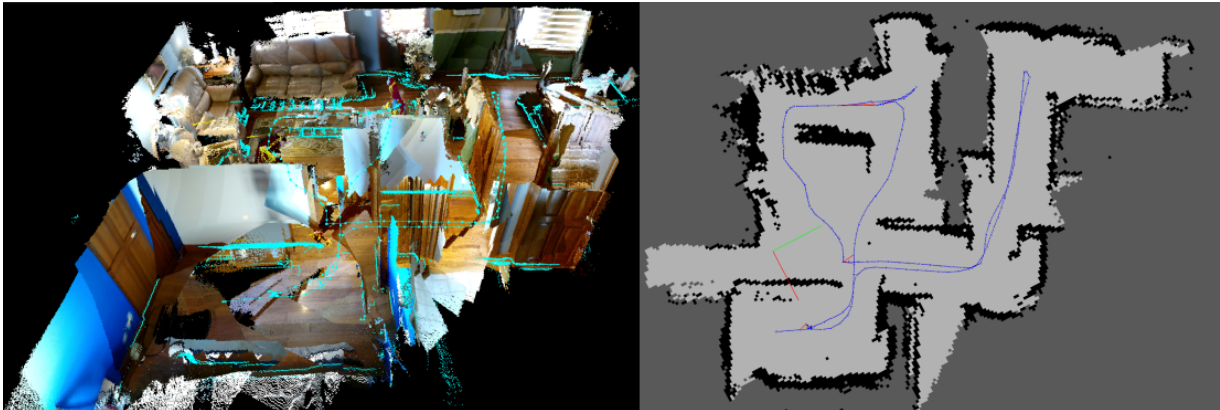


Figure 3.2 Map generated by RTAB-Map

### 3.5.2 ODAS

Similarly to artificial vision, artificial audition consists of the ability to derive auditory information about the environment from the robot. Robots for home assistance would operate in noisy environments, and limitations are observed in such conditions when using only one or two microphones [71]. Using a microphone array can enhance performance by allowing a robot to localize, track, and separate multiple sound sources to improve situation awareness. ODAS [20] is our newest open source library<sup>4</sup> performing sound sources localization, tracking and separation. ODAS is derived from our previous work on ManyEars [60], also distributed as an open source library<sup>5</sup>. ManyEars uses the generalized cross-correlation phase transform for audio localization, and a particle filter based method for tracking. Released in 2009, ManyEars has been downloaded more than 8355 times. ManyEars was used in December 2009 by the winning team *Fly By Ear* of the Annual Machine Intelligence Competition, run by the British Computer Society.

Figure 3.3 shows the main components inside the ODAS framework. ODAS improves robustness to noise by allowing to increase the number of microphones used while reducing computational load. This library relies on a new localization method called Steered Response Power with Phase Transform based on Hierarchical Search with Directivity model and Automatic calibra-

4. <http://odas.io>

5. <https://github.com/introlab/manyyears>

tion (SRP-PHAT-HSDA). Localization generates noisy potential sources, which are then filtered with a novel tracking method based on a modified 3D Kalman filter (M3K) that generates one or many tracked sources. The module's output can be used to continuously orient the robot's heading in the speaker's direction, and sound locations can be displayed on the remote operator 3D interface [72]. Sound sources are then filtered and separated using directive geometric source separation (DGSS) to focus the robot's attention only on speech, and ignore ambient noise. This new library also models microphones as sensors with a directive polar pattern, which improves sound sources localization, tracking and separation when the direct path between microphones and the sound sources is obstructed by the robot's body. Figure 3.4 presents the ODAS Studio application<sup>6</sup> displaying real time 3D sound source locations for fine tuning of microphone positions and ODAS parameters.

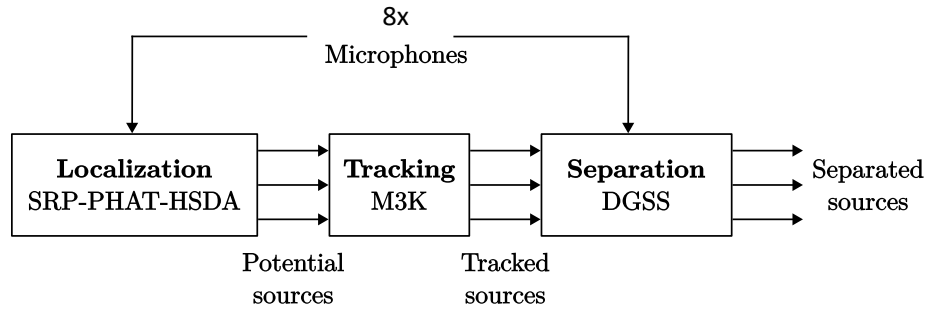


Figure 3.3 ODAS architecture

To make use of ODAS or ManyEars, a sound card and microphones are required. Commercial sound cards present limitations when used for embedded robotic applications : they are usually expensive ; they have functionalities such as sound effects, integrated mixing, optical inputs/outputs, S/PDIF, MIDI, numerous analogs outputs, etc., which are not required for artificial audition ; they also require significant amount of power and size. To facilitate the use of ODAS and ManyEars on various robotic platforms, we also provide as open hardware two sound cards : 8SoundsUSB<sup>7</sup> and 16SoundsUSB<sup>8</sup> (released in January 2018), for eight and sixteen microphone arrays, respectively. They provide synchronous acquisition of microphone signals through USB to the robot's computer. The 8SoundsUSB kit has been downloaded more than 8850 times, and is used by research labs around the world [73, 74, 75].

### 3.5.3 HBBA

Design of an interactive mobile robot is one [76] if not the most challenging integration problem in robotics. It involves dealing with action (manipulation, mobility), perception (environment,

6. [https://github.com/introlab/odas\\_web](https://github.com/introlab/odas_web)

7. <https://sourceforge.net/projects/8soundsusb/>

8. <https://github.com/introlab/16SoundsUSB>

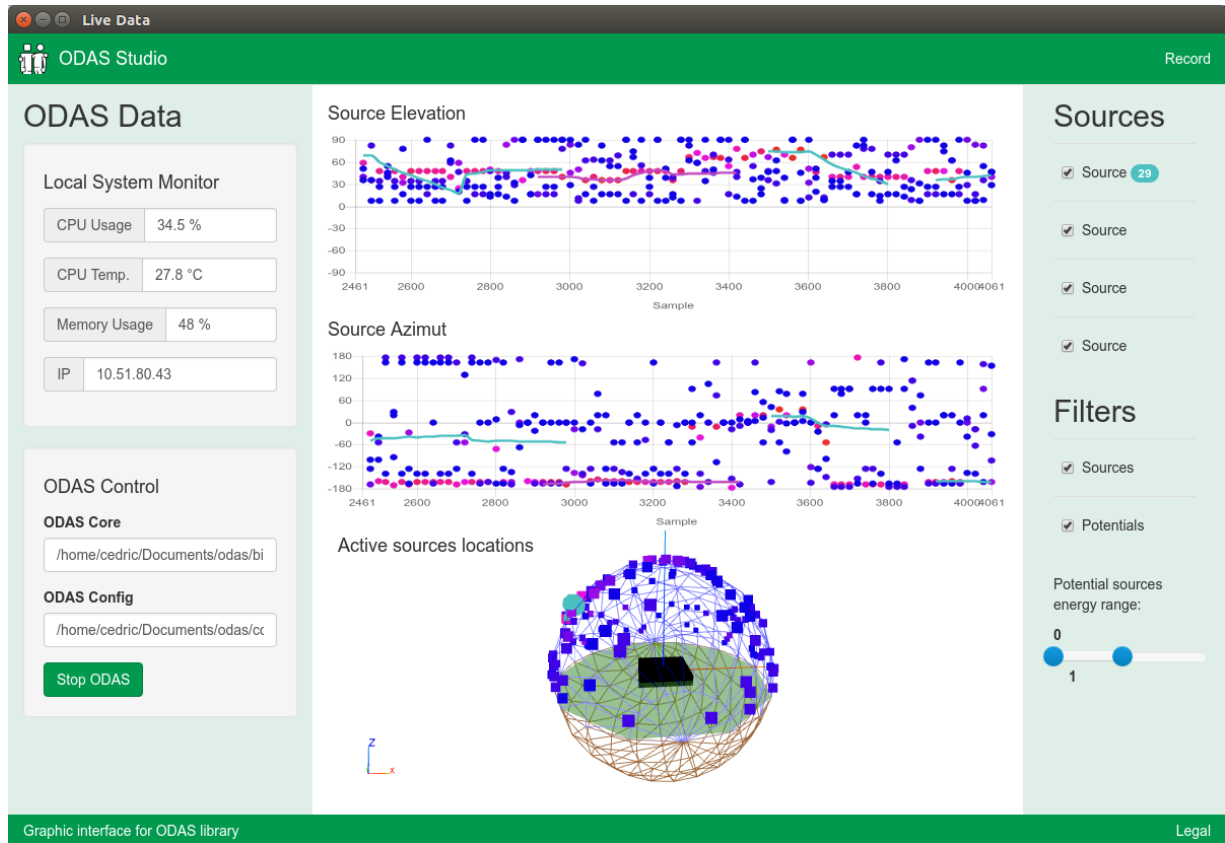


Figure 3.4 ODAS studio application

people), interaction (information exchange modalities such as interpretation of perceptual cues, human-robot interfaces, etc.), systems (mechatronics, control, software, cognition) [77], in relation to an application domain. These elements are all interdependent, as each one influences the others.

To address this challenge, robot control architectures define the interrelations between decision-making modules required by the application. There is an infinite number of ways to implement robot control architectures, and although there is no consensus on a common architecture, how to engineer a system that effectively integrates the functionalities required is an open question of fundamental importance in robotics and human-robot interaction (HRI) [78]. Integration and coordination of different types of processing modules (perception, reasoning, behaviors) is required, and there is currently no dominant solution [79].

A frequently used control architecture is the layered, or tiered, robot control architecture, with layers usually organized according to the principle of increasing precision with decreasing intelligence [80]. For instance, the Donaxi robot [45] has a deliberative (for symbolic representation and reasoning), an executive (for plan monitoring) and a functional layer. Siepmann et al. [46]

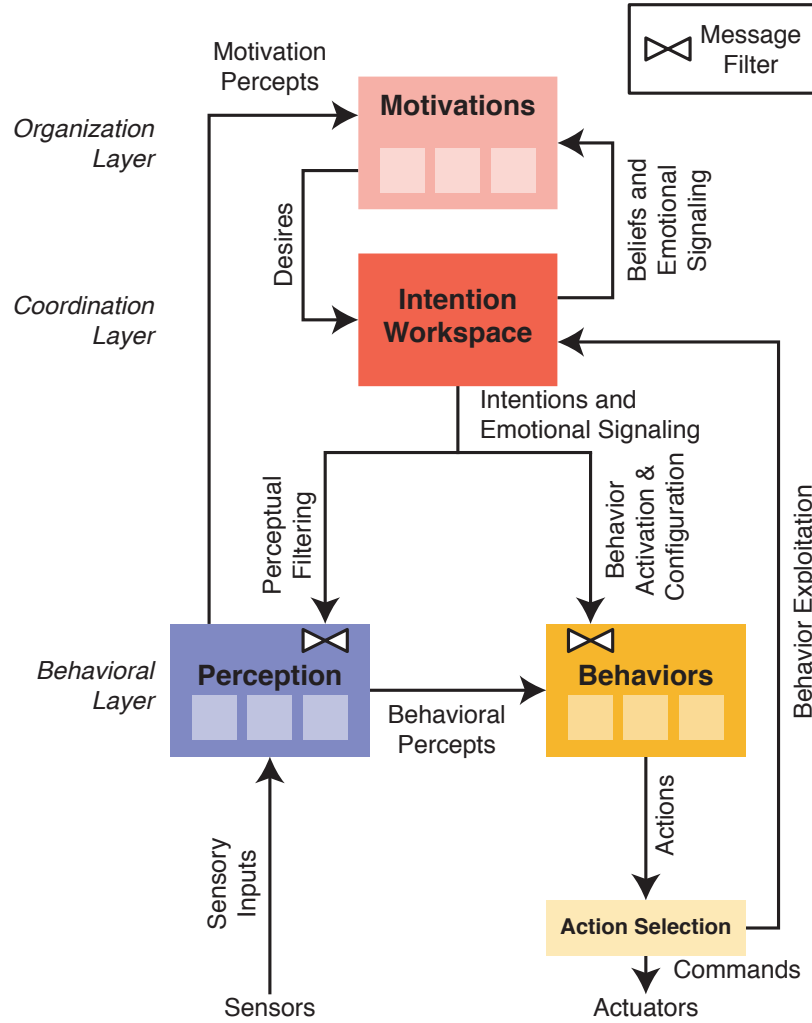


Figure 3.5 Hybrid Behavior-Based Architecture (HBBA)

uses a hardware, a functional and a BonSAI layer. The complexity in layered robot control architecture comes in how to interface and partition these layers [81]. To address these issues, we use HBBA (Hybrid Behavior-Based Architecture) [61, 17], an open source<sup>9</sup> and unifying framework for integrated design of HRI scenarios. Illustrated by Fig. 3.5, HBBA is a behavior-based architecture with no central representation that provides the possibility of high-level modeling, reasoning and planning capabilities through Motivations or Perception modules. Basically, it allows Behaviors to be configured and activated according to what are referred to as the Intentions of the robot. Intentions are structured data providing the configuration and activation of Behaviors (i.e., the behavioral strategy) and the modulation of Perception modules. As the number and complexity of Perception modules, Behaviors and Motivations increase to address more sophisticated interaction scenarios, the Intention Workspace becomes critical. While lay-

9. <http://github.com/francoisferland/hbba>



red architectures usually impose a specific deliberative structure (for instance a task planner) to coordinate the lower-level Behaviors, HBBA allows multiple concurrent independent modules at its highest level, without constraining those modules to a specific decisional scheme. Compared to more formal planning approaches such as Konidaris and Hayes [82], HBBA is a robot control architecture presenting design guidelines and working principles for the different processing modules, without imposing a formal coding structure for its implementation. HBBA's generic coordination mechanism of Behaviors has demonstrated its ability to address a wide range of cognitive capabilities, ranging from assisted teleoperation to selective attention and episodic memory, simply by coordinating the activation and configuration of perception and behavior modules. It also has been used with humanoid robots such as the NAO and Meka Robotics M1 in a episodic memory sharing setup [83], and with the Robosoft Kompai and later on the PAL Robotics TIA Go as service robots for the elderly with mild cognitive impairments [84].

## 3.6 SAM's robot control architecture

Figure 3.6 illustrates the implementation of SAM's robot control architecture, following the HBBA framework, to make SAM a remote assistance robot. To give a general overview, its main motivations are Survive and Assistive Teleoperation. Survive supervises the battery level and generates a Desire to go to the charge station when battery level is too low. Using the interface, the remote operator can activate autonomous functionalities managed by Assistive Teleoperation. This allows the user to either manually control the robot, to communicate a high level destination for autonomous navigation, to follow autonomously a conversation or to autonomously orient SAM towards a person. The following sections provide more details on the Sensors, Perception, Behaviors, Actuators and Motivations modules implemented for SAM.

### 3.6.1 Sensors

SAM has the following input sensory modules :

- **Battery Level** monitors the battery voltage level and current consumption in floating point units.
  - **Gamepad** is a wireless controller used to activate or deactivate the wheel motors. It allows the operator to manually navigate the robot or to activate modes from the Operator GUI.
  - **Operator GUI** (General User Interface) shown in Fig. 3.7 can be used by the operator in teleoperation mode.
  - **Kinect** is the data generated by the RGB-D camera.
  - **Floor Camera** is a webcam facing the ground and used to locate the charge station.
-

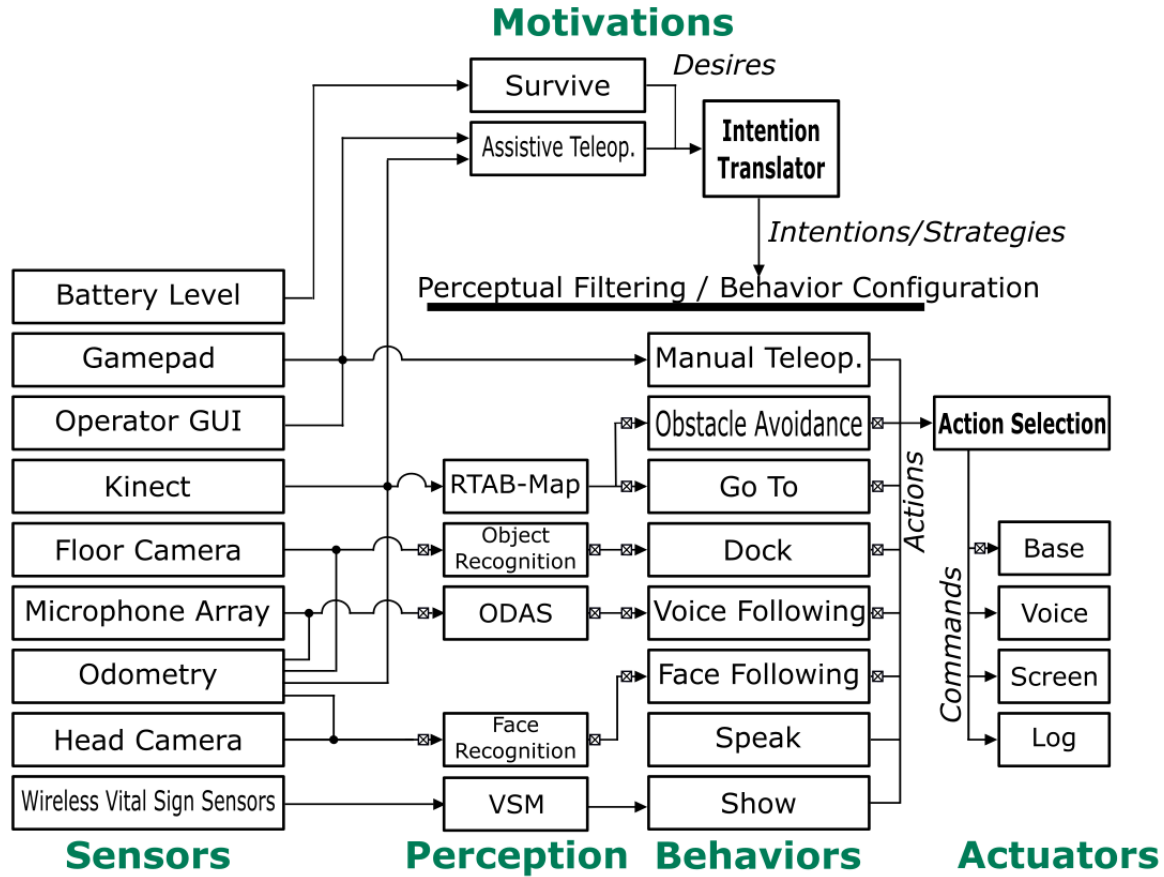


Figure 3.6 SAM's robot control architecture using HBBA

- **Microphone Array** is the 8-microphone array installed on SAM.
- **Odometry** is data provided by wheel encoders and the inertial measurement unit of the Beam+ platform to estimate the change of position of the robot over time.
- **Head Camera** is the webcam installed in SAM's forehead, facing forward, for visual interaction with people.
- **Wireless Vital Sign Sensors** is the BLE interface for the wireless vital signs monitoring devices.

## Perception

The Perception modules translate Sensors data into useful information for the Behaviors. SAM's Perception modules are :

- **RTAB-MAP** for mapping and localization, as presented in Section 3.5.1.

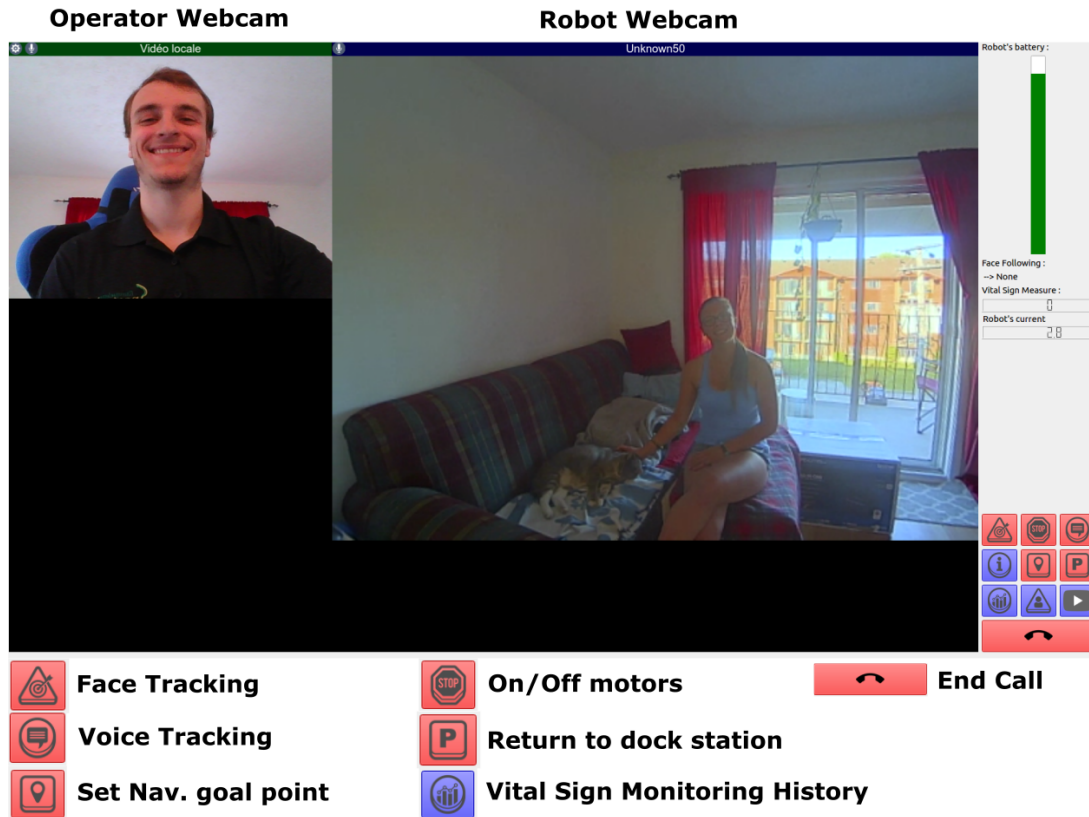


Figure 3.7 Teleoperation GUI

- **Object Recognition** uses Floor Camera to detect the charging station and identifies its orientation using ROS find\_object\_2d package<sup>10</sup>.
- **ODAS** is for sound source localization, tracking and separation, as explained in Section 3.5.2.
- **Face Recognition** uses SAM's Head Camera in conjunction with FisherFaceRecognizer<sup>11</sup> from the OpenCV 3.0 library. It compares the actual face with all the prerecorded faces in its database. Then, Face Recognition identifies the most likely person with a confidence score.
- **VSM** (Vital Sign Monitoring) translates vital sign data into a generic JSON format, to be compatible with software used by Show and Log.

## Behaviors

SAM's Behaviors are control modalities organized with a priority-based action selection scheme as follows :

10. [http://wiki.ros.org/find\\_object\\_2d](http://wiki.ros.org/find_object_2d)

11. [https://docs.opencv.org/3.0.0/da/d60/tutorial\\_face\\_main.html](https://docs.opencv.org/3.0.0/da/d60/tutorial_face_main.html)

- **Manual Teleoperation** is the highest priority Behavior, giving absolute control to an operator using the Gamepad. This Behavior is used for security interventions and during a mapping session.
- **Obstacle Avoidance** plans a path around an obstacle detected in the robot's local map, to avoid collisions.
- **Go To** allows SAM to navigate autonomously using SPLAM provided by RTAB-MAP.

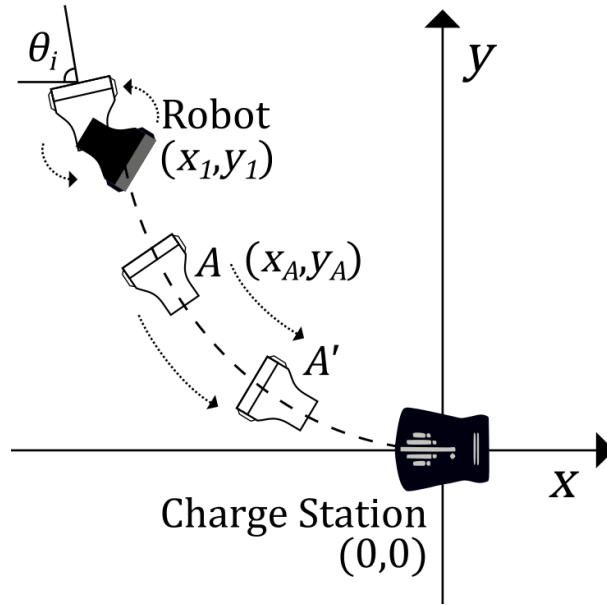


Figure 3.8 2D representation of the path generated by the Dock behavior

- **Dock** allows the robot to connect itself to the charge station when it is detected. With the Beam's head's hard drive replaced, we could not interface this behavior with the existing docking algorithm of the Beam+ platform. Therefore, we had to implement our own behavior. The robot charging connector is at the back of its base and there is no sensor to navigate backwards. Therefore, before turning to dock backwards, the robot must generate a path. Shown in Fig. 3.8, our algorithm uses a  $(x,y)$  representation centered on the charge station at  $(0,0)$ , pointing to the left  $(-x)$  and the robot position at  $(x_1, y_1)$ . Object Recognition gives the position  $(x_R^S, y_R^S)$  and orientation  $\theta_R^S$  of the charge station relative to the robot. Then the robot position relative to the charge station  $(x_1, y_1)$  is found :

$$x_1 = -x_R^S \cos \theta_R^S - y_R^S \sin \theta_R^S \quad (3.1)$$

$$y_1 = x_R^S \sin \theta_R^S - y_R^S \cos \theta_R^S \quad (3.2)$$

To connect the robot perpendicularly to the charge station, a 2nd order polynomial path is chosen :

$$y = \frac{y_1}{x_1^2} x^2 \quad (3.3)$$

$$(3.4)$$

Once the path is calculated, the initial orientation  $\theta_i$  of the robot is found using the derivative of (3.3). The robot turns in place to reach  $\theta_i$ . It then starts to move backwards following the path. To monitor the movement, Odometry provides the robot position  $(x_M^A, y_M^A)$  and orientation  $(\theta_M^A)$  with respect to the map. This position in relation to the charge station  $(x_A, y_A)$  is found by applying the homogeneous matrices  $(A_R^M, A_S^R)$  that change the coordinate system from map to robot and from robot to charge station, respectively.

$$\begin{bmatrix} x_A \\ y_A \\ 1 \end{bmatrix} = A_R^M A_S^R \begin{bmatrix} x_M^A \\ y_M^A \\ 1 \end{bmatrix} \quad (3.5)$$

$$(3.6)$$

Using a cycling rate of 100 Hz, the velocities are defined by a translational velocity of 0.4 m/s and a rotational velocity defined by :

$$\dot{\theta}_{twist} = (\theta_{A'} - \theta_A) \times 100 \quad (3.7)$$

When Odometry indicates that the robot is not moving and that there is indeed a non-zero speed command sent to the base, it means that the robot encountered an obstacle, potentially the charging station. It then stops for 1 sec and if the battery's current consumption becomes negative within this period, the robot is docked and charging. If not, the robot resumes the trajectory.

- **Voice Following** uses ODAS to perceive multiple sound source locations, amplitudes and types (voice or non-voice). The main interlocutor is considered to be the voice source with the highest energy. Its location and the robot's odometry are used to turn and face the main interlocutor.
- **Face Following** follows the closest face detected by Face Recognition using the TLD Predator (Tracking Learning and Detection) [85] package. As shown by Fig. 3.9, once a face is detected, Face Following is able to track it even if it becomes covered or it changes orientation. The current implementation only tracks one face at a time.

- **Speak** converts pre-defined texts into speech using Festival speech synthesis and the sound\_play ROS package <sup>12</sup>.
- **Show** displays, on the robot's screen, the remote operator webcam, vital signs and the robot's battery level, as shown by Fig. 3.7.

### Actuators

The Action Selection module receives all the Actions generated by the activated Behaviors and keeps the ones from the highest priority Behaviors for the same Actuator. Actuators are :

- **Base** translates velocity commands into control data for the wheel motors.
- **Voice** plays sounds coming from Speak or the audio coming from the operator's voice.
- **Screen** displays the info from Show.
- **Log** saves all vital signs gathered from VSM into a Firebase database, a Google web application <sup>13</sup>. Data are logged with a time stamp.

### Motivations

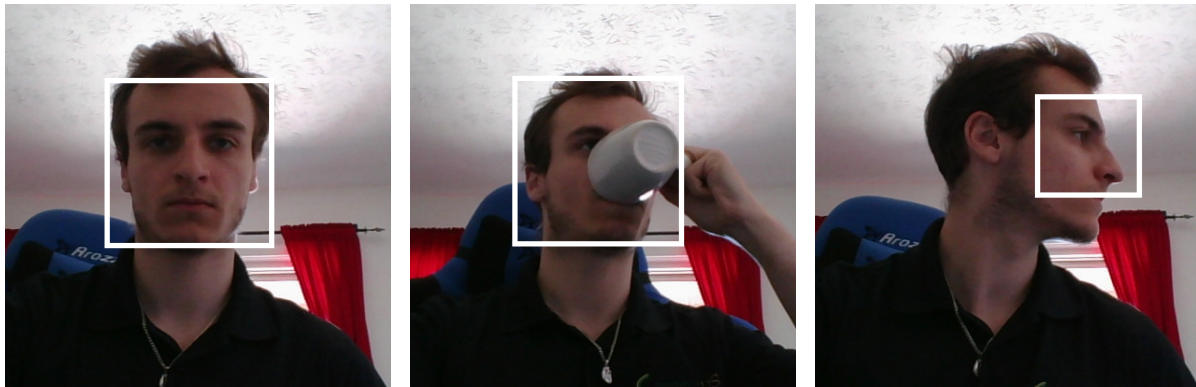
SAM's Motivations are :

- **Survive** monitors SAM's Battery Level and generates a Desire to return to the charging station when battery voltage is lower than 11.5 V.
- **Assistive Teleoperation** allows the remote operator, using the GUI to activate autonomous modes for navigation, for following a conversation or for following a person's face.

---

12. [http://wiki.ros.org/sound\\_play](http://wiki.ros.org/sound_play)

13. <https://firebase.google.com/>



(a) Face detected

(b) Covered face

(c) Changed orientation

Figure 3.9 Face Following capabilities

SAM allows the operator to map the environment, to navigate autonomously in the resulting map, to follow a conversation and to follow a face. These modalities can be individually activated using Operator GUI. Before conducting trials with typical users in home care scenarios, we decided to first characterize the robot's performance of two of its more specific functionalities, i.e., the ability to navigate autonomously in the homes, and the ability to autonomously follow a conversation, which are associated to our open source libraries. Navigation was tested in our lab facility. SAM safely moved in hallways around people and in the lab around workbenches and tables, shelves and other robots. The motivation Survive, i.e., making the robot return to the charge station, was successfully validated over traveling distances ranging from 1 to 20 m. To simulate real home conditions, we placed an obstacle against the wall to lower the door frame width to 71 cm. Autonomous conversation following was tested in different rooms and during public demonstrations. Face recognition was validated with different participants individually, also in different rooms.

We then tested SAM's capabilities in 10 real home environments listed in Table 3.4, with a variety of rooms, furniture, floor type, doors width and hallways width. None were modified or adjusted to help the robot, except for doors that were either fully opened or closed. The charging station was positioned against a wall in an area with enough space for the robot to turn and dock. Every door frames, door thresholds and hallways crossed were measured.

### 3.6.2 Autonomous navigation

Tableau 3.1 Navigation results from 10 home environments

	Number of Trials	Succeeded with Intervention	Succeeded Autonomously	Intervention to Prevent Collision	Intervention to Help Loop Closure	Intervention to Help Dock the Robot
Navigate in a Room	97	94 (97%)	81 (84%)	11 (11%)	5 (5%)	0 (0%)
Navigate to a Different Room	173	170 (98%)	126 (73%)	36 (21%)	18 (10%)	0 (0%)
Return to the Charging Station Located in the Same Room	51	39 (76%)	33 (65%)	1 (2%)	3 (6%)	19 (37%)
Return to the Charging Station Located in the Different Room	79	65 (82%)	45 (57%)	16 (20%)	11 (14%)	25 (32%)
Total	400	368 (92%)	288 (72%)	64 (16%)	37 (9%)	44 (11%)

Table 3.1 presents the results of 400 maneuvers in different conditions, using 35 different maps made in the 10 homes. In all trials, SAM tried to achieve the maneuvers autonomously, but in some cases the operator had to intervene. The operator used the GUI to look at the representation

of the robot in its map and to watch the images from both the head and floor cameras. The operator used the Gamepad to intervene by navigating the robot to map the environment, to avoid collision if navigation failed to produce a successful path around an obstacle or to help the robot overcome a path planning failure. To avoid collisions, the operator stopped the robot and changed its orientation to move away from the obstacle, to then let the robot resume autonomous navigation. Operator intervention occurred 64 times during 54 trials to avoid a collision, 37 times over 30 trials to help detect loop closure when path planning fails, and 44 times in 33 trials to help dock to the charge station. For recharging, when SAM was unsuccessful in docking to the charging station, the operator manually positioned the robot in front of the charge station and reset the command. If it happened twice, the attempt was considered unsuccessful. During a trial, the operator could intervene more than once and the trial was considered to have failed if at least one intervention was unsuccessful. In such conditions, SAM completed a total 288 (72%) maneuvers autonomously and 368 (92%) commands were successful with operator intervention.

Distance travelled by the robot to reach its goal point was between 3.4 to 11.9 m with an average of 6.9 m. As shown in Fig. 3.10, the robot took 14 to 158 sec with an average of 38.5 sec to reach its goal point. Each operator intervention added time to reach the goal point, which explains the cases over 54 sec.

Figure 3.11 shows the environment A, B, E and J, respectively, to provide examples of typical representations made using RTAB-Map. The sketches are approximate representations of the real homes, and RTAB-MAP illustrates the resulting map. The dark lines in the maps are obstacles and the grey areas are the safe zones for navigation.

SAM had the best performance when navigating inside a room. Random goal points were chosen and repeated at least 3 times with the same initial condition. If SAM had the same issue twice at the same location, the attempt was considered unsuccessful. Limiting factors for SAM navigation capabilities are, in order of impact :

- **SAM's odometry.** SAM Odometry is calculated by the Beam+ base using wheel encoders and an inertial measurement unit. Rotation error is around 2.8% and linear error is roughly 0.8%. For each rotation in place, the odometry accumulates an error of up to 10°, which decreases the quality of the map derived by RTAB-Map.
- **Limited field of view (FOV), as illustrated by Fig. 3.12.** To deal with the blind spot during the mapping sessions, the operator made the robot stay at least 40 cm away from obstacles that could be partially unperceivable because of the blind spot. However, in autonomous navigation, if the robot went too close to a counter, a coffee table or a chair, the local map did not properly show the obstacle, thus increasing the risk of collisions.



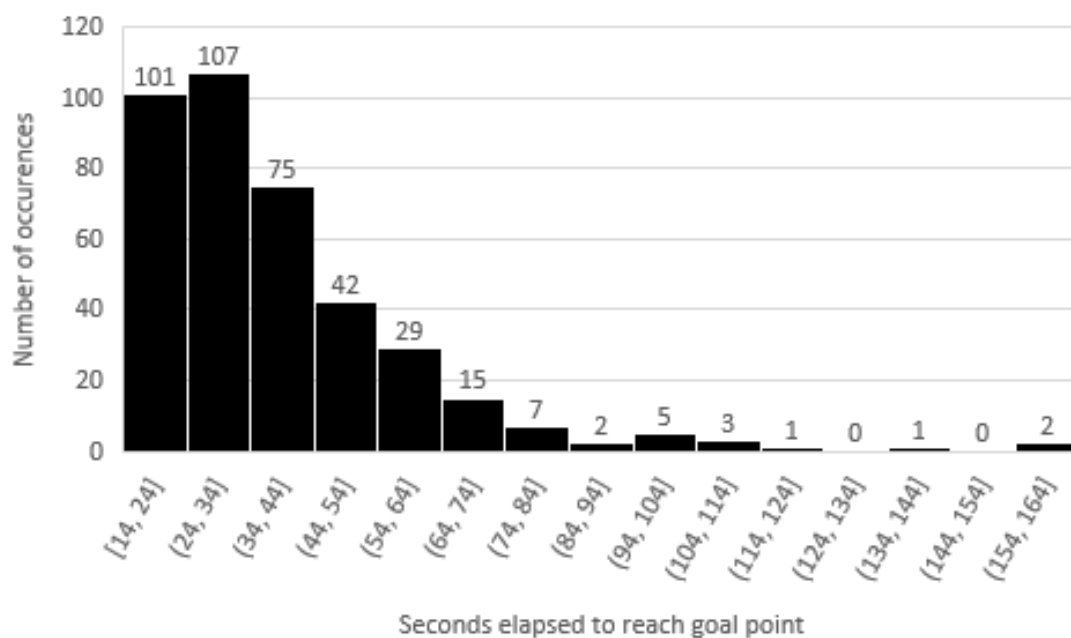


Figure 3.10 Navigation time histogram

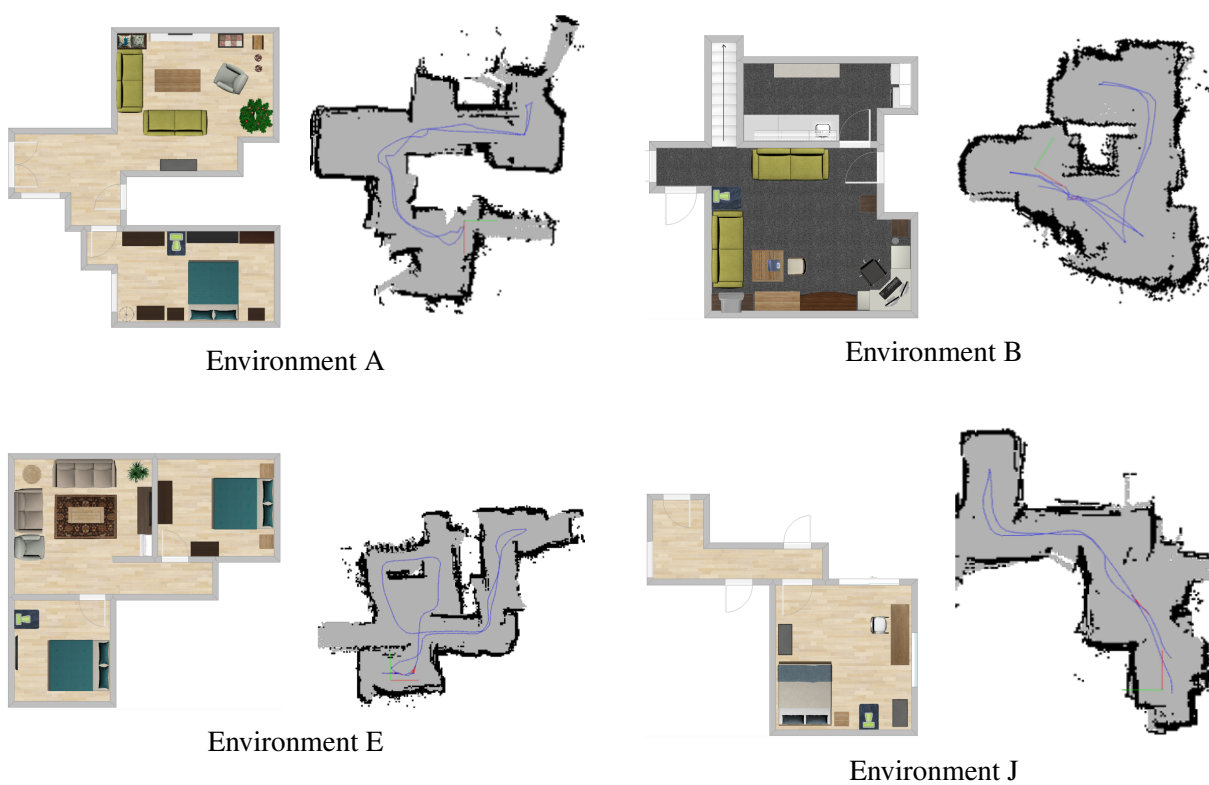


Figure 3.11 2D representation of the Environment A, B, E and J (left : Sketch, right : RTAB-Map)

This limitation is the cause of the 11 and 36 interventions to prevent collisions while navigating in a room compared to navigating to a different room, respectively.

- **Change or lack of visual features for loop closure detection.** When RTAB-Map is unable to perform loop closure, the odometry error accumulates and the local map drifts from the global map. If the drift becomes too large, RTAB-Map is unable to find a possible path to both satisfy the local map and the global map, making navigation impossible. In this situation, the operator had to manually navigate the robot until RTAB-Map could perform a loop closure and resynchronized SAM in the map. If this resynchronization did not occur in less than 30 sec, the attempt was considered unsuccessful. In the 37 interventions made by the operator, only 2 times it resulted in an unsuccessful attempt to perform a loop closure.
- **Vibration.** With the Kinect placed on the Beam+ top screen, vibration becomes a source of errors while mapping and localizing. As shown in Fig. 3.13, a simple 2° oscillation can cause misinterpretation errors, for instance registering the floor as an obstacle. Robot's accelerations, floor slope and floor cracks cause such oscillations. To prevent this, we set the minimum object height at 20 cm. Although this solution eliminates floor misinterpretation, small objects on the floor such as shoes or toys are not seen as obstacles. For example, in environment D, the robot planned a path that could have hit a walker's wheel if the operator did not intervene.
- **Reflective objects.** The Kinect camera has problems sensing reflective objects such as mirrors. All obstacles reflected by a mirror are seen as through a window. When mapping, this may add noise or ghost obstacles in other rooms. To minimize this effect, rooms with large mirrors should be mapped first and then, every room affected by the reflection should be mapped afterwards.

SAM navigating to a different room also revealed difficulties, mostly occurring when crossing a door frame. Table 3.2 presents observations over 287 door frames crossed in relation to door width. Senior residences, with 83 cm width door frame adapted for wheelchairs, show the best performance, while anything smaller resulted in similar success rate. Tight spaces, narrow turns and furniture near door frames were almost nonexistent in these senior residences, decreasing the occurrences of having an obstacle in the robot's blind spot and helping to find a valid path despite a drift in the local map.

SAM also experienced difficulties planning a path towards the goal point because of inabilities to detect loop closures. When mapping, RTAB-Map stores images with visual features as references for loop closure. One limitation is that every features are assumed static and significant. This caused problems in laundry rooms and kitchens. For example, the top room of Environment

---

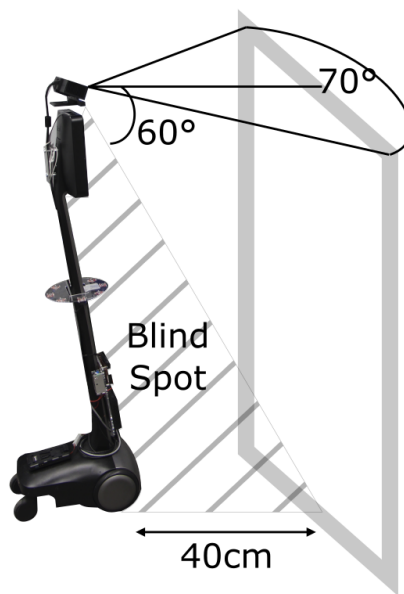


Figure 3.12 Kinect's blind spot on SAM

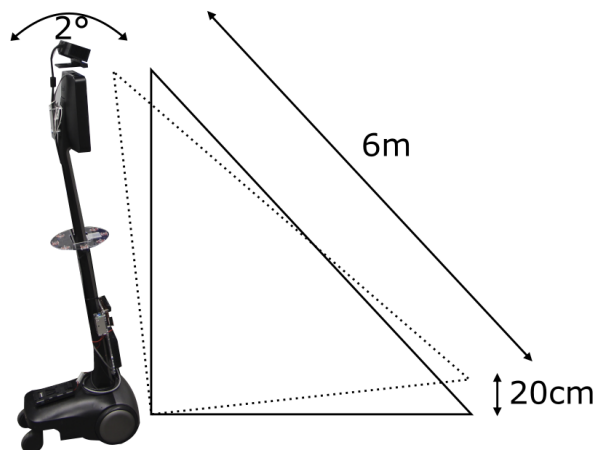


Figure 3.13 Floor misinterpretation caused by vibration

Tableau 3.2 Autonomous Navigation Through Door Frames

Door Width	Attempts	Succeeded Autonomously
58 cm	24	79%
71 cm	132	78%
76 cm	85	77%
83 cm	46	98%
Total	287	81%

B in Figure A.2 is a laundry room. The first time it was mapped, it had colorful clothes folded on the ironing table. The next day, RTAB-Map was unable to perform loop closure because the clothes were gone and the colorful features were not visible. This problem can occur with all kinds of objects in the homes such as dishes, food, shoes, clothes, pets and even doors (opened or closed), chairs or plants. To avoid this, mapping should be done continuously and proximity sensors (e.g., lidar) should be added for improved robustness, or visual object classification should be done to remove potentially unreliable visual features.

Referring back to Table 3.1, returning to the charging station, whether placed in the same room or in another room, also revealed some limitations. Returning to the charging station combines the use of Go To with the saved location for the charging station, Object Recognition to detect the charging station, and Dock to back up on the charging station. As shown in Fig. 3.14, the charge station has a flat part to the ground that represents a symbol used by Object Recognition to calculate the distance and orientation of the charging station. Depending on illumination conditions, the symbol may not be defined enough and the orientation estimate can have an error of up to  $20^\circ$ . Also, the flat part of the charging station is made of metal, which has low friction. If SAM's propelling wheels move from a high friction surface (e.g., carpet, anti-slip lenoleum) to a low friction surface, the wheels may spin for a short time because the motor controller temporarily overshoots the amount of power sent to the motors. This makes the robot deviate from its planned trajectory, making it unable to dock correctly.

### 3.6.3 Autonomous conversation following

Autonomous conversation following aims to enhance the operator experience by autonomously direct the camera towards the person talking. To provide repeatable experimental conditions, a pre-recorded audio conversation between two men were played from two speakers placed at different heights (43 cm to 1.4 m), angles ( $120^\circ$  to  $150^\circ$  and distances (1 m to 1.6 m) in



Figure 3.14 SAM's charging station

environments A, B, E and J. from Table 3.4. We consider that an interlocutor change occurs when the person speaks for more than 4 syllables, which happened 12 times in the test.

Table 3.3 presents the observed results. Quiet condition consists of no loud interference heard during the test, during which SAM succeeded 93% of the time, and in the remaining 7% the robot remained still.

Noisy conditions consists of doing the same test but with typical sounds that can occur in home settings : home residents were told to resume their normal activities, and therefore could watch television, listen to music, prepare meals, vacuum, etc., in addition to regular home noise (e.g., kitchen hood, fan). Performance then dropped to 62%. Interfering sound sources which included voices, such as television and music lyrics, were sometimes considered as a valid interlocutor, making the robot turn towards it. On the other hand, kitchen hood and vacuum cleaner noise were very rarely detected as voice. This is explained by ODAS' voice detection algorithm, which analyzes the frequency range of the sound source : if the interfering sound overlapped in the 85 to 255 Hz interval, false recognition may occur.

## 3.7 Conclusion

This paper presents our implementation of SAM, semi-autonomous telepresence mobile robot for remote home care assistance. It outlines the different elements that come into play, from navigation to interaction modalities and their hardware, software and decision-making integration. It is still quite a challenge to improve reliability and robustness of the different modalities, and our experiments in real home settings revealed the capabilities and limitations of the platform. These limitations are important because they will certainly influence results from usability studies : either the operating environments are constrained or engineered in some ways, or the robot platform is changed to minimize the occurrences of those limitations. Obviously the robot still needs to be teleoperated to be used in home settings, and full autonomy could be considered as a future feature, making evaluation be done by comparing complete manual control with semi to complete autonomy, always keeping the robot under the surveillance of the operator (similarly to current autonomous driving vehicles).

Tableau 3.3 Autonomous conversation following

Scenario	Quiet	Noisy
Change of Interlocutor	288	252
Followed Successfully	93%	62%

In spite of those limitations, SAM performed reasonably well in real home settings, and we learned a lot from conducting trials in unconstrained conditions, pointing out interesting issues to improve. In our future work, we will be focusing more our effort on senior residences. Following a user-centered design approach involving clinicians, seniors and caregivers, we plan to conduct trials to illustrate what can be done with the robot to help identify interaction scenarios to be conducted with SAM. We also plan to expand and port what was implemented with SAM on other robot platforms with improved sensing and interaction capabilities.

## Acknowledgement

This work is supported by AGE-WELL, the Canadian Network of Centres of Excellence on Aging Gracefully across Environments using Technology to Support Wellness, Engagement, and Long Life.

Tableau 3.4 Description of the 10 Real Home Tested

	Home Type	Floor Type	Door Frame Width	Hallway Width	Door Threshold Height	Rooms	Furniture
A	Apartment	Hardwood	71 cm			Bedroom and Living Room	Bed, Nightstand, Dressers, Mirror, Sofas, Coffee Tables and Christmas Tree
B	Basement	Rough Carpet	76 cm			Laundry Room and Living Room	Ironing Table, Counters, Washer, Dryer, Desk, Sofas, Chair, Table and Filing Cabinet
C	Ground Floor	Hardwood	71 cm	-	0.3 cm	Living Room, Entrance Room, Bathroom, Dinning Room and Kitchen	Sofas, Coffee Table, TV, Cat Tree, Counter, Mirror, Toilet, Washer, Dryer, Table, Chairs and Counters
D	Senior Residence	Laminate	76 cm, 83 cm	1.1 m	0.3 cm	Bedroom, Living Room, Bathroom, Kitchen and Hallway	Bed, Dresser, Walker, Armchairs, Tables, TV, Chairs, Counters, Bath, Toilet and Mirror
E	Ground Floor	Hardwood	71 cm	86 cm	0.4 cm	Living room, Bedrooms, Hallway	Sofas, Rug, Coffee Table, Plant, Bed, Dresser, Nightstand and Bed
F	Senior Residence	Anti-Slip Lino-leum	83 cm			Living Room, Bedroom, Kitchen and Bathroom	Armchairs, TV, Coffee Table, Bed, Dresser, Counter, Table, Chairs, Counter, Bath and Toilet
G	Senior Residence	Anti-Slip Lino-leum	83 cm			Living Room, Office, Kitchen, Dining Room and Bedroom	Armchairs, TV, Coffee Table, Desk, Chairs, Mirror, Counter, Table, Bed and Dresser
H	Ground Floor	Laminate				Living Room, Empty Room	Sofas, Armchair and Round Table
I	Basement	Rough Carpet	58 cm, 71 cm, 76 cm	112 cm		Laundry room, Walk-ins, Training room and Hallway	Counter, Ironing Table, Shelves and Training Equipment
J	Apartment	Laminate	76 cm	86 cm	0.3 cm	Entrance Room, Bedroom and Hallway	Bed, Nightstand, Desk, Chair, Dresser and Mirror





# CHAPITRE 4

## Assistance autonome à la prise de mesure des signes vitaux

Individuellement, les fonctionnalités offertes via la motivation *Assistive Teleoperation* visent à améliorer l'expérience de téléprésence de l'opérateur en facilitant la navigation du robot et les interactions avec l'utilisateur. Ces fonctionnalités peuvent aussi être utilisées dans un contexte entièrement autonome. Dans le but de voir jusqu'où il est possible de se rendre dans l'exploitation des capacités du robot, nous avons mis en place la motivation AVSM (*Autonomous Vital Sign Monitoring*), dans la configuration montrée à la figure 4.1, afin d'étudier ce qui est envisageable de réaliser avec la plateforme SAM dans un scénario entièrement autonome. Le scénario se décrit comme suit :

1. Le robot se réveille à un temps défini par l'opérateur.
2. Le robot navigue dans la maison à l'aide d'une liste pré-enregistrée de points d'intérêt dans la carte de l'environnement (*RTAB-MAP* et *Go To*).
3. À l'arrivée de chaque point d'intérêt, le robot parle et demande à l'utilisateur de venir prendre ses mesures de signes vitaux (*Speak*).
4. L'utilisateur peut répondre verbalement ou se présenter face au robot :
  - Si l'utilisateur parle, le robot détecte la voix et s'oriente vers lui (*ODAS* et *Voice Following*).
  - Lorsque le visage est détecté, le robot s'assure d'identifier que c'est la personne d'intérêt que l'opérateur a demandé pour faire le suivi des signes vitaux (*Face Recognition* et *Face Following*).
5. Une fois l'identification approuvée, le robot demande les mesures de signes vitaux requis (*Speak*).
6. Après chaque mesure, le robot enregistre dans sa base de donnée la date, la personne, l'instrument de mesure et la valeur mesurée (*VSM* et *Log*).
7. Lorsque toutes les mesures sont effectuées, le robot remercie l'utilisateur (*Speak*) et retourne à sa station de recharge (*RTAB-MAP* et *Go To*).
8. À proximité de la station de recharge, le robot l'identifie et se stationne (*Object Recognition* et *Dock*).

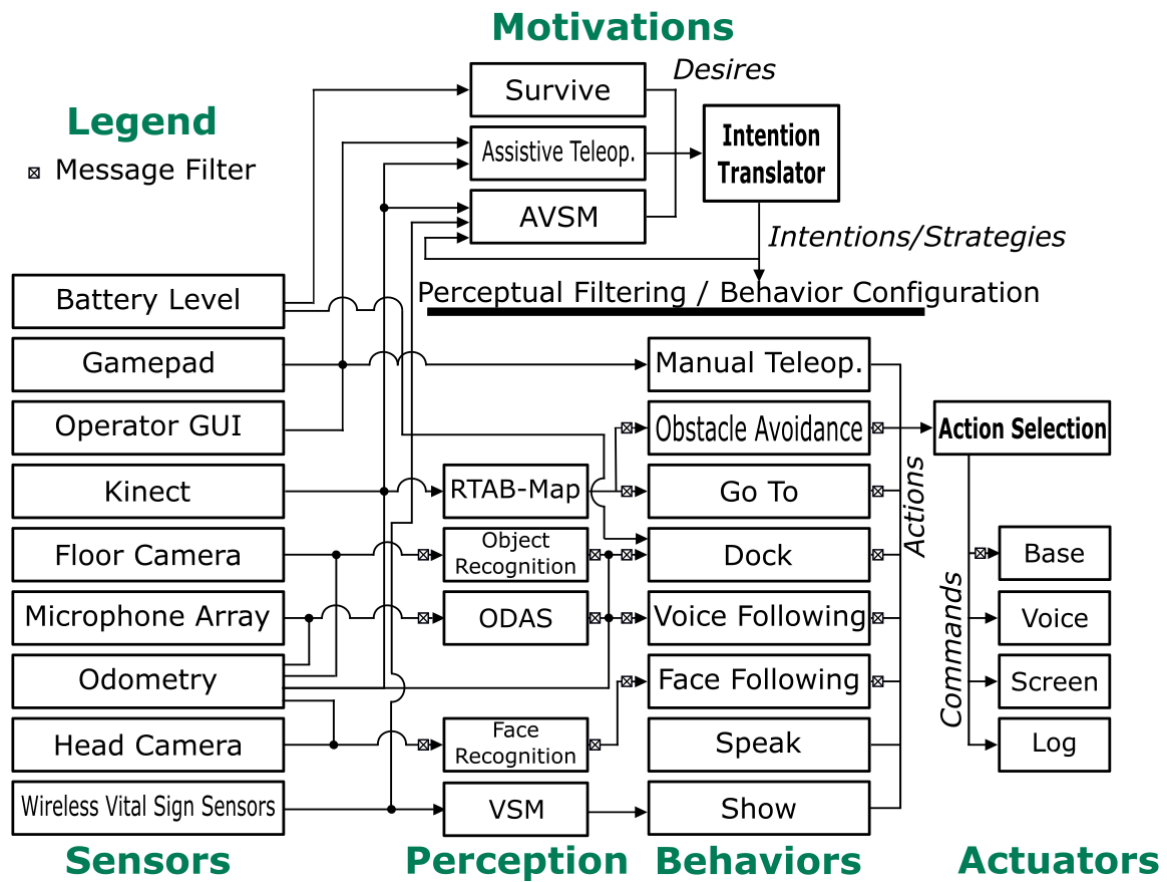


Figure 4.1 Ajout de la motivation AVSM à l'architecture de contrôle de SAM

La motivation AVSM tente ainsi d'utiliser le plein potentiel de toutes les modalités implémentées dans la configuration actuelle de HBBA. Toutefois, l'évaluation préliminaire d'AVSM a résulté en un taux de réussite de l'ordre de 15%. Le robot partait de sa station de recharge pour changer de pièce, demandait l'utilisateur, se tournait vers sa voix lorsque ce dernier lui répondait, identifiait son visage, demandait les mesures de signes vitaux et retournait à sa station de recharge et se stationnait de reculons. Le robot a réussi à accomplir ce scénario de façon autonome seulement 2 fois sur les 13 essais. Cela peut s'expliquer par le fait que les performances d'AVSM combinent celles des fonctionnalités individuelles présentées à la section 3.6. Théoriquement, le taux de réussite  $TR$  d'AVSM se traduit par la formule suivante, avec  $N$  étant le nombre de *GoTo* à réaliser dans le cadre d'une intervention :

$$TR_{AVSM} = (TR_{Navigate\_in\_a\_room})^N \times (TR_{Navigate\_to\_a\_different\_room})^N \times TR_{Voice\ Following} \times TR_{Dock} \quad (4.1)$$

Pour ces essais, l'algorithme de reconnaissance faciale ne fut pas utilisé car l'algorithme compare le visage actuel avec ceux enregistrés dans sa base de données. Un visage est considéré détecté lorsque l'algorithme fournit un taux de confiance supérieur à un seuil prédéfini. Si le robot fait face à un nouveau visage, celui-ci sera associé à un visage enregistré. De plus, l'angle du visage, son évolution au travers le temps (cheveux, pilosité, accessoires) et la luminosité jouent tous un rôle important sur le taux de confiance.

Considérant les performances présentées au tableau 3.1 et au tableau 3.3, les taux de réussite théoriques d'AVSM dépendent de la complexité des tâches à accomplir ainsi que de la possibilité d'intervention par un opérateur. Si nous calculons le taux de réussite de l'évaluation préliminaire réalisée, le robot devait naviguer à l'intérieur d'une même pièce, pour ensuite aller à deux autres pièces (sans considérer les largeurs de porte), suivre une conversation dans un environnement calme et retourner à sa borne de recharge en repassant à travers les deux pièces. Au final, les performances théoriques résultantes seraient :

$$TR_{AVSM\_sans\_intervention} = (84\%)^1 \times (73\%)^4 \times 93\% \times 57\% = 13\% \quad (4.2)$$

$$TR_{AVSM\_avec\_intervention} = (97\%)^1 \times (98\%)^4 \times 100\% \times 82\% = 73\% \quad (4.3)$$

Le scénario réalisé par la motivation AVSM s'avère donc envisageable lorsque l'opérateur peut intervenir au besoin, mais reste encore trop embryonnaire pour être réalisée en mode entièrement autonome.



# CHAPITRE 5

## Conclusion

Lors de cette étude, des fonctionnalités de navigation, de suivi de la voix, de suivis du visage et d'acquisition d'informations ont été implémentées sur la plateforme robotique Beam+. Cette implémentation est réalisée par l'unification des bibliothèques RTAB-Map et ODAS dans l'architecture décisionnelle HBBA. Cette solution a été caractérisée dans 10 différents domiciles qui ont permis de mettre en évidence les défis et les limites d'un tel déploiement.

Les résultats de la navigation démontrent sa viabilité lorsque l'opérateur la supervise. Cependant, sans intervention, les capacités restent insuffisantes pour assurer la robustess requise. Lorsque utilisé dans les domiciles adaptés à l'usage de fauteuil roulant, la navigation est avanta-gée par l'environnement plus dégagé et son taux de réussite en est favorisé. La position, l'angle, la qualité et les limitations des capteurs jouent un rôle dominant sur la qualité de la carte de référence et la capacité à s'y repositionner. Un des critères du choix du matériel ajouté était de garder le prix du robot le plus bas possible. Par contre, il serait intéressant de caractériser de nouveau le robot dans les domiciles avec des capteurs beaucoup plus précis, tel que le Google Tango, pour quantifier l'impact au niveau du taux de réussite des navigations sans supervision. Les résultats du suivi de conversation démontre qu'en environnement calme, l'opérateur pourrait facilement suivre un dialogue entre deux utilisateurs sans avoir à intervenir. En milieu bruyant, il serait potentiellement possible d'améliorer les résultats avec l'usage d'un plus grand nombre de microphones.

Pour la suite des travaux, il serait intéressant d'acquérir une autre plateforme robotique afin de voir comment compenser face aux limitations observées sur la mise en œuvre de SAM sur Beam+. Une nouvelle configuration de capteurs composée d'une caméra stéréo pour la navi-gation, d'un lidar, d'une matrice de 16 microphones et d'une caméra HD pour la détection du visage pourraient améliorer les performances. Il faudra également évaluer l'état de l'art des algo-rithmes de reconnaissance faciale afin d'être plus précis et robuste. Avec l'architecture HBBA, il sera aisément possible de procéder à ce transfert. Finalement, nous croyons qu'à moyen terme, le robot pourra être évalué par des professionnels de la santé. Des groupes de discussion et des essais en milieux réels nous permettront d'en apprendre davantage sur les requis des utilisateurs finaux.



## ANNEXE A

### Caractéristiques des 10 domiciles

Tableau A.1 Description des domiciles où furent réalisés les tests

ID	Type de domicile	Type de plancher	Largeur porte (cm)	Largeur couloir (cm)	Hauteur seuil de porte (mm)	Pièces	Meubles	Fig
A	Appartement	Bois franc	71	-	-	(a) Chambre (b) Salon	(a) Lit, table de chevet, commode, miroir (b) Sofas, table à café, arbre de Noël	A.1
B	Sous-sol	Tapis court	76	-	-	(a) Buanderie (b) Salon	(a) Table à repasser, comptoir, laveuse, sècheuse (b) Bureau, sofas, chaise, table, classeur	A.2
C	Rez-de-chaussée	Bois franc	71	-	3	(a) Salon (b) Entrée (c) Salle de bain (d) Salle à manger (e) Cuisine	(a) Sofas, table à café, TV, arbre à chat (b) Vide (c) Comptoirs, miroir, toilette, laveuse, sècheuse (d) Table, chaises (e) Comptoirs	A.3
D	Résidence pour personnes âgées	Plancher flottant	76 83	110	3	(a) Chambre (b) Salon (c) Salle de bain (d) Cuisine (h) Couloir	(a) Lit, commode, marchette, chaise (b) Fauteuil, tables, TV, chaises (c) Comptoirs, bain, toilette, miroir (d) Comptoirs (h) Vide	A.4
E	Rez-de-chaussée	Bois franc	71	86	4	(a) Salon (b) Chambre (c) Chambre (h) Couloir	(a) Sofas, tapis, table à café, plante (b) Lit, commode, table de chevet (c) lit (h) Vide	A.5
F	Résidence pour personnes âgées	Linoléum antidérapant	83	-	-	(a) Salon (b) Chambre (c) Cuisine (d) Salle de bain	(a) Fauteuil, TV, table à café (b) Lit, commode (c) Comptoir, table, chaises (d) Comptoir, bain, toilette	A.6
G	Résidence pour personnes âgées	Linoléum antidérapant	83	-	-	(a) Salon (b) Bureau (c) Cuisine (d) Salle à manger (e) Chambre	(a) Fauteuil, TV, table à café (b) Bureau, chaises, miroir (c) Comptoirs, chaises (d) Table, chaises (e) Lit, commode	A.7
H	Rez-de-chaussée	Plancher flottant	-	-	-	(a) Salon (b) Pièce vide	(a) Sofas, fauteuils, table ronde (b) Vide	A.8
I	Sous-sol	Tapis court	58 71 76	112	-	(a) Buanderie (b) Dépense (c) Salle d'entraînement (h) Couloir	(a) Comptoir, table à repasser (b) Étagères (c) Équipement d'entraînement (h) Vide	A.9
J	Appartement	Plancher flottant	76	86	3	(a) Entrée (b) Chambre (h) Couloir	(a) Vide (b) Lit, table de chevet, bureau, chaise, commode, miroir	A.10



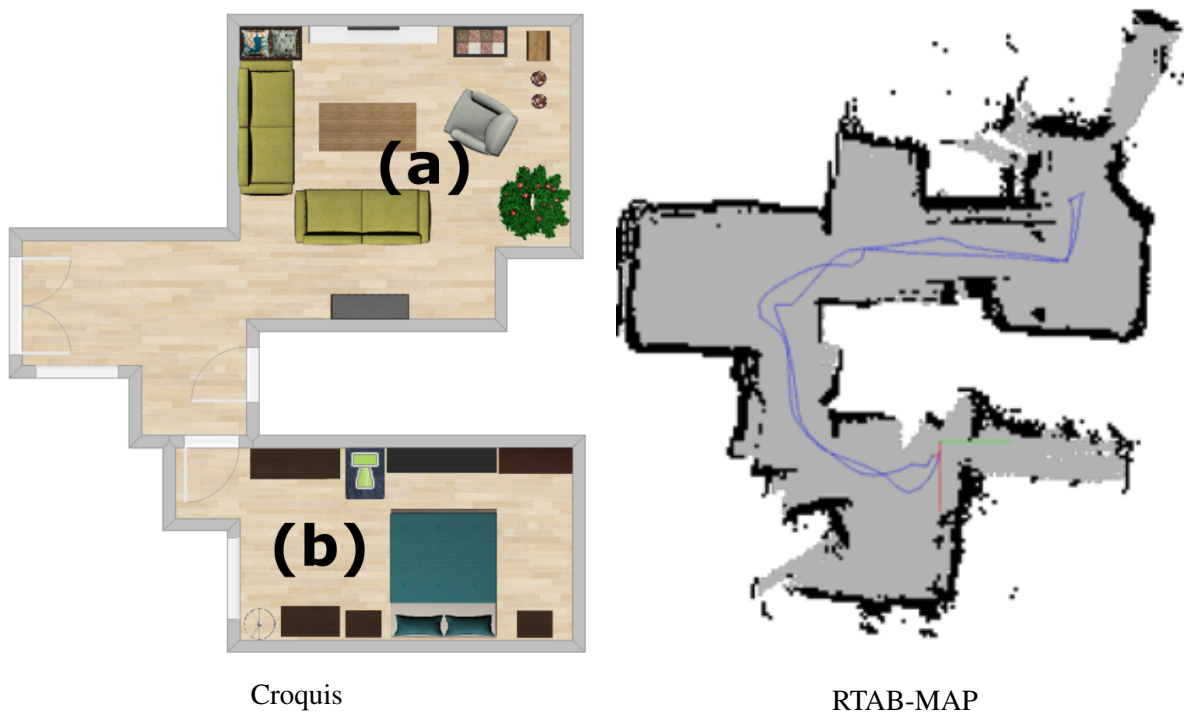


Figure A.1 Environnement A

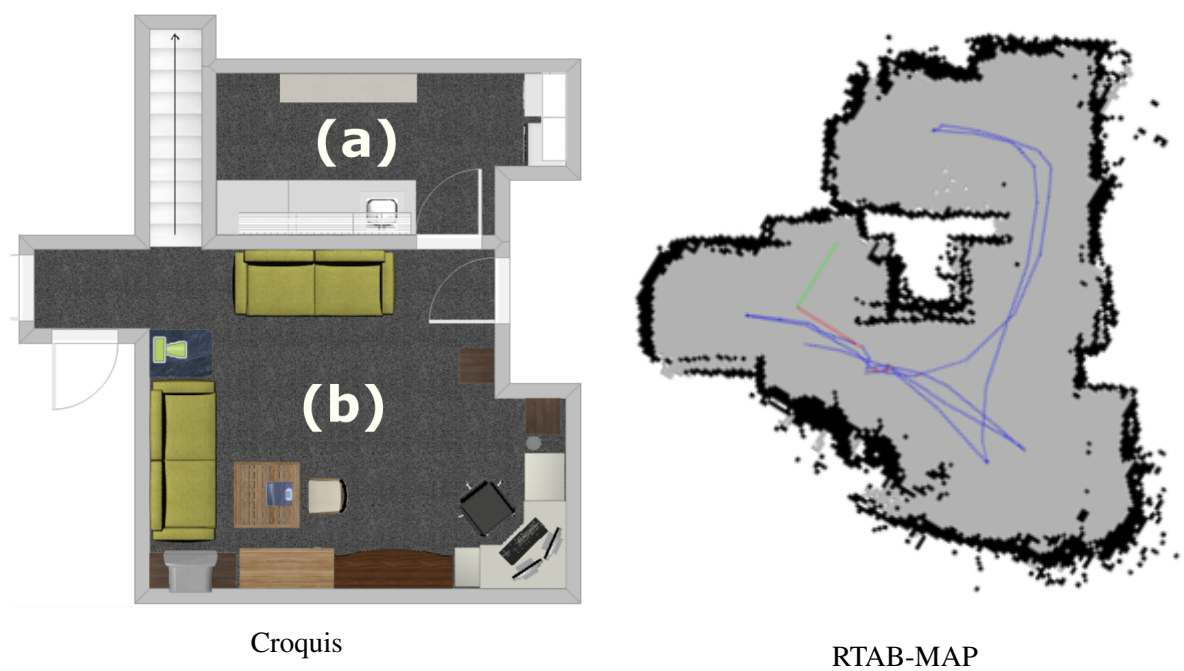
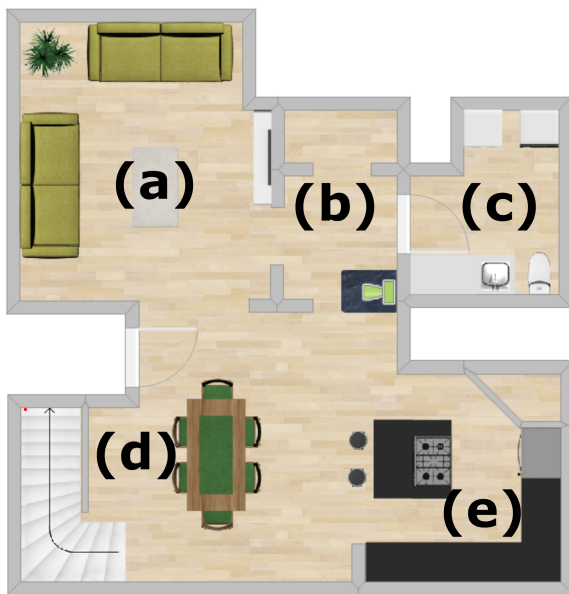
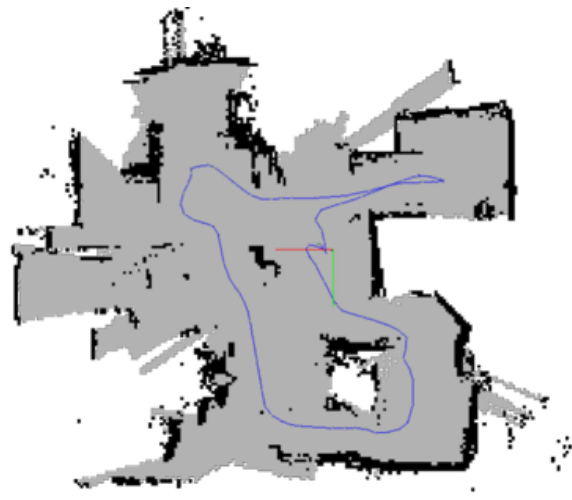


Figure A.2 Environnement B

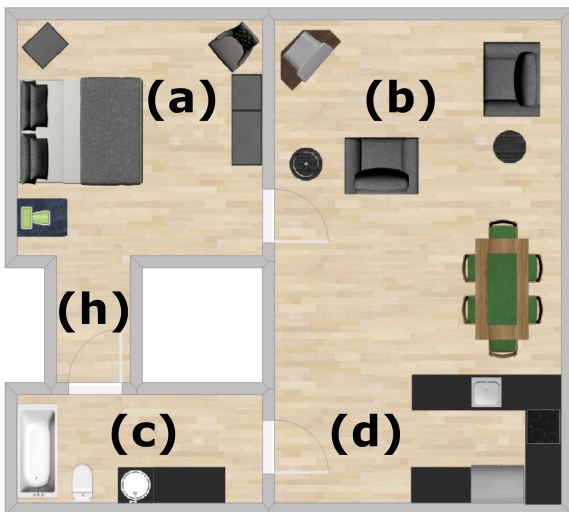


Croquis

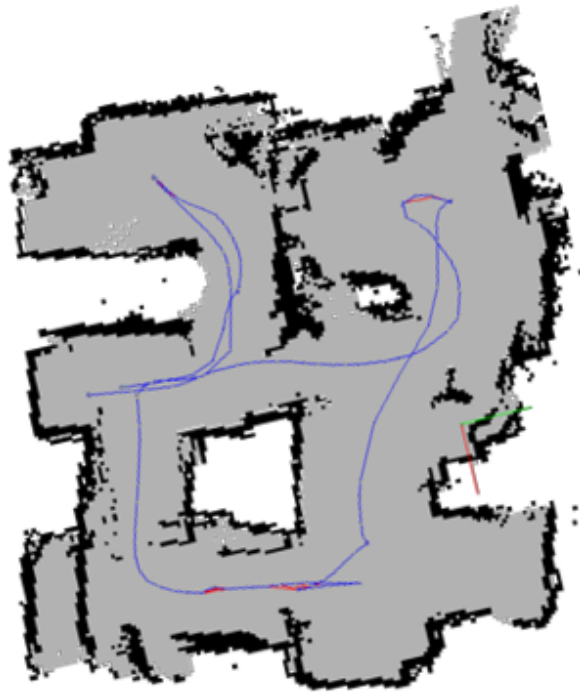


RTAB-MAP

Figure A.3 Environnement C



Croquis



RTAB-MAP

Figure A.4 Environnement D

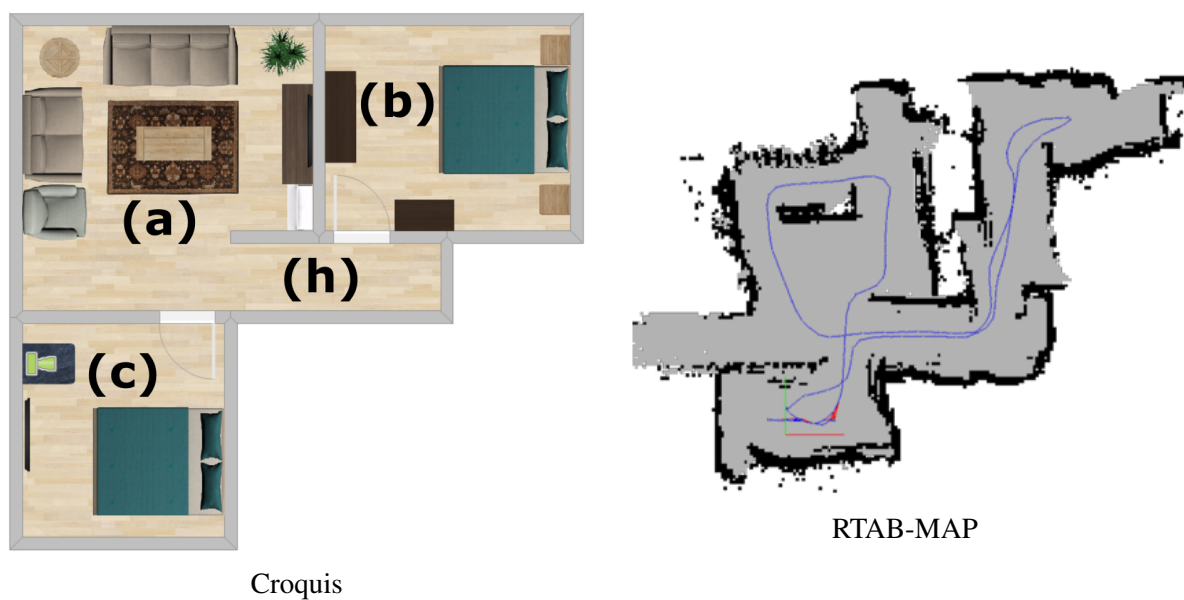


Figure A.5 Environnement E

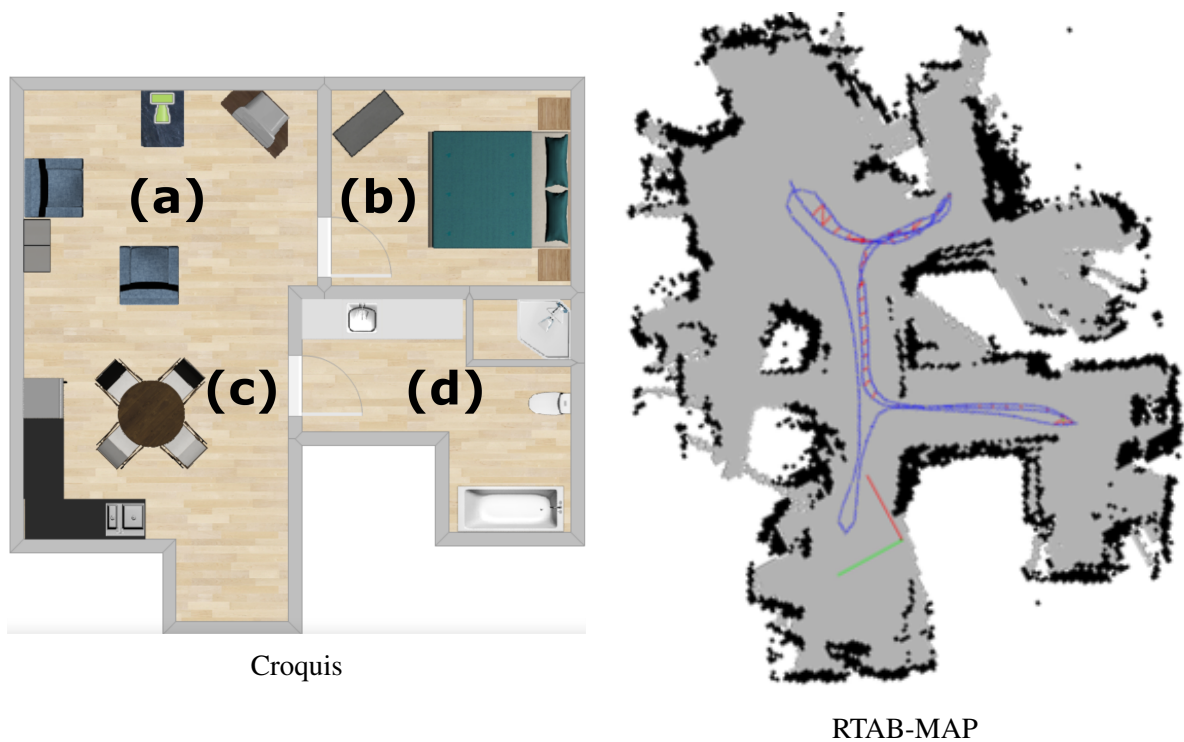
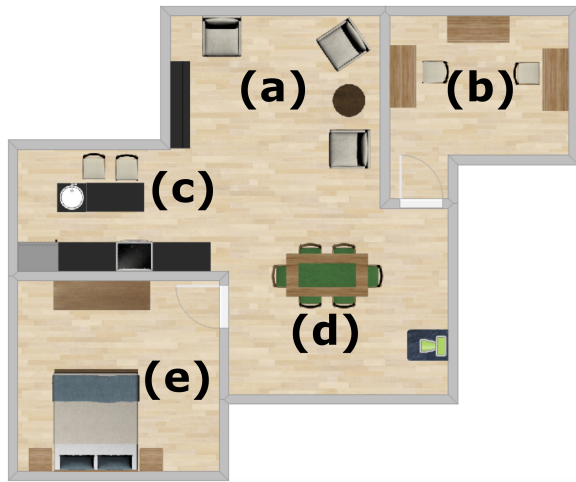


Figure A.6 Environnement F

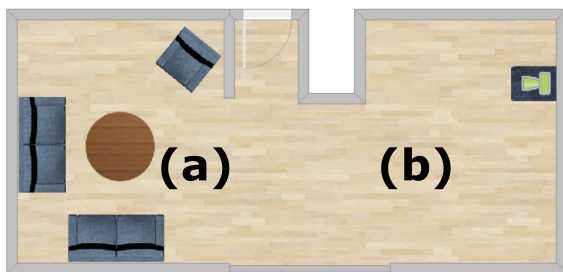


Croquis



RTAB-MAP

Figure A.7 Environnement G



Croquis



RTAB-MAP

Figure A.8 Environnement H

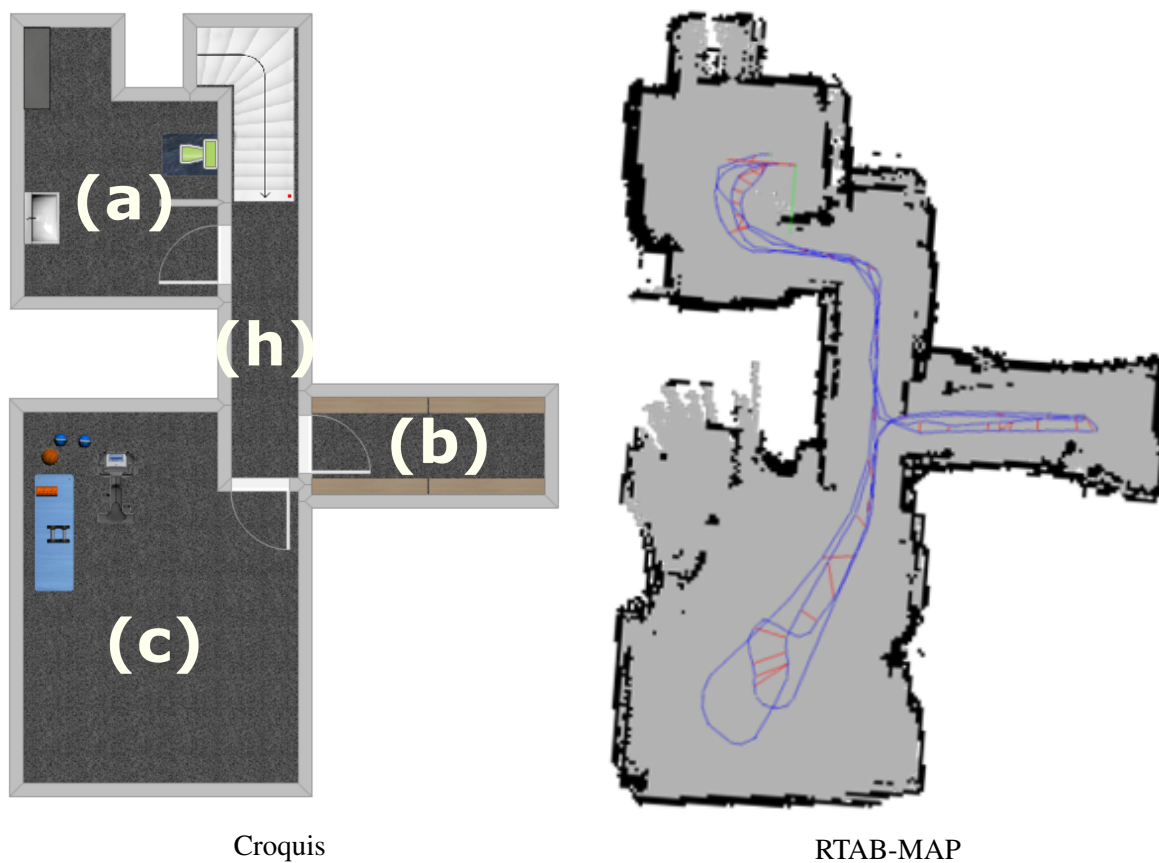


Figure A.9 Environnement I

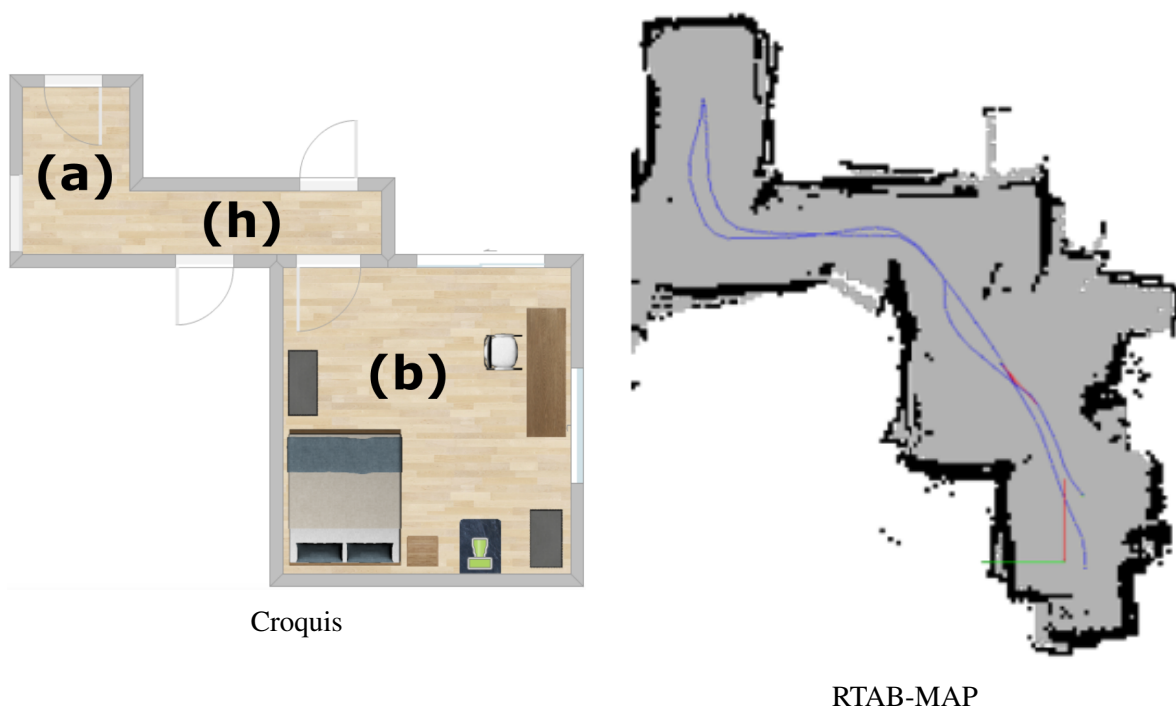


Figure A.10 Environnement J





## ANNEXE B

### Analyse du comportement *Dock*

Le comportement *Dock* permet au robot de se connecter par lui-même à la station de recharge. Avec le disque dur de la tête du Beam+ remplacé, nous n'avons plus accès à l'algorithme de stationnement utilisé par le robot Beam+ de base. Par conséquent, nous avons dû concevoir notre propre algorithme.

Le connecteur du robot se situe à l'arrière de sa base et il n'y a aucun capteur pour reculer. Ainsi, avant de se retourner, le robot doit générer sa trajectoire. Illustré à la figure 3.8, notre algorithme utilise une représentation  $(x,y)$  centrée à la station de recharge  $(0,0)$ , face à gauche  $(-x)$  et la position du robot à  $(x_1, y_1)$ . *Object Recognition* donne la position  $p_R^S$  et l'orientation  $\theta_R^S$  de la station de recharge par rapport au robot. La position du robot relativement à la station de recharge  $(x_1, y_1)$  est trouvée en appliquant la transformation homogène  $A_S^R$ , qui transforme la référence du robot vers la station de recharge :

$$\tilde{p}_S^R = \begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix} \quad (\text{B.1})$$

$$\tilde{p}_S^R = A_S^R \tilde{p}_R^R \quad (\text{B.2})$$

$$\tilde{p}_R^R = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad (\text{B.3})$$

$$A_S^R = \begin{bmatrix} R_R^{S^T} & -R_R^{S^T} p_R^S \\ 0_{1 \times 2} & 1 \end{bmatrix} \quad (\text{B.4})$$

$$R_R^S = \begin{bmatrix} \cos \theta_R^S & -\sin \theta_R^S \\ \sin \theta_R^S & \cos \theta_R^S \end{bmatrix} \quad (\text{B.5})$$

$$p_R^S = \begin{bmatrix} x_R^S \\ y_R^S \end{bmatrix} \quad (\text{B.6})$$

$$x_1 = -x_R^S \cos \theta_R^S - y_R^S \sin \theta_R^S \quad (\text{B.7})$$

$$y_1 = x_R^S \sin \theta_R^S - y_R^S \cos \theta_R^S \quad (\text{B.8})$$

Pour se stationner perpendiculairement à la borne de recharge, une équation de second degré est utilisée :

$$y = ax^2 \quad (\text{B.9})$$

$$a = \frac{y_1}{x_1^2} \quad (\text{B.10})$$

$$y = \frac{y_1}{x_1^2} x^2 \quad (\text{B.11})$$

Une fois l'équation obtenue, l'orientation initiale  $\theta_i$  du robot est obtenue avec la dérivé de (B.9) :

$$\theta = \tan^{-1}\left(\frac{dy}{dx}\right) \quad (\text{B.12})$$

$$\frac{dy}{dx} = 2ax \quad (\text{B.13})$$

$$\theta = \tan^{-1}\left(\frac{2y_1}{x_1^2}x\right) \quad (\text{B.14})$$

$$\theta_i = \tan^{-1}\left(\frac{2y_1}{x_1^2}(x_1)\right) \quad (\text{B.15})$$

$$\theta_i = \tan^{-1}\left(\frac{2y_1}{x_1}\right) \quad (\text{B.16})$$

$$(\text{B.17})$$

Le robot tourne sur lui-même pour atteindre  $\theta_i$ . Par la suite, il commence à reculer en suivant la trajectoire polynomiale à l'aide de l'odométrie qui fournit  $p_M^A$ ,  $\theta_M^A$ , la position et l'orientation du robot par rapport à sa carte. Pour les exprimer par rapport à station de recharge, la transformation homogène  $A_S^M$  est utilisée :

$$\tilde{p}_S^A = A_S^M \tilde{p}_M^A \quad (\text{B.18})$$

$$\tilde{p}_S^A = \begin{bmatrix} x_A \\ y_A \\ 1 \end{bmatrix} \quad (\text{B.19})$$

$$A_S^M = A_R^M A_S^R \quad (\text{B.20})$$

$$\tilde{p}_M^A = \begin{bmatrix} x_M^A \\ y_M^A \\ 1 \end{bmatrix} \quad (\text{B.21})$$

$$A_R^M = \begin{bmatrix} R_M^{R^T} & -R_M^{R^T} p_M^R \\ 0_{1 \times 2} & 1 \end{bmatrix} \quad (\text{B.22})$$

$$R_M^R = \begin{bmatrix} \cos \theta_M^R & -\sin \theta_M^R \\ \sin \theta_M^R & \cos \theta_M^R \end{bmatrix} \quad (\text{B.23})$$

Avec un taux de rafraîchissement de 100 Hz, les vitesses sont définies par une vitesse linéaire constante à 0.4 m/s et une vitesse en rotation de :

$$\dot{\theta}_{twist} = (\theta_{A'} - \theta_A) \times 100 \quad (\text{B.24})$$

Lorsque l'odométrie indique qu'il n'y a plus de mouvement malgré qu'il a une commande non nulle envoyée à la base, cela veut dire que le robot a heurté un obstacle, potentiellement la station de recharge. Dans cette circonstance, le robot s'arrête une seconde et surveille la consommation de sa batterie. Si elle devient négative, cela veut dire que le robot est stationné et se recharge. Sinon, le robot reprend la poursuite de sa trajectoire.



# LISTE DES RÉFÉRENCES

- [1] R. Schraft, “The concept of a system for assisting elderly or disabled persons in home environments,” *Proceedings of the 24th IEEE IECON*, 1998, vol. 4, pp. 183–190, 1998.
- [2] “Statistique canada - une population vieillissante.” <https://doi.org/10.25318/1710000501-eng>. Consulté en Mai 2019.
- [3] F. G. Miskelly, “Assistive technology in elderly care,” *Age and ageing*, vol. 30, no. 6, pp. 455–458, 2001.
- [4] S. Ghidoni, S. M. Anzalone, M. Munaro, S. Michieletto, E. Menegatti, “A distributed perception infrastructure for robot assisted living,” *Robotics and Autonomous Systems*, vol. 62, no. 9, pp. 1316–1328, 2014.
- [5] M. Nadi, N. El-Bendary, H. Mahmoud, A. E. Hassanien, “Fall detection system of elderly people based on integral image and histogram of oriented gradient feature,” *2014 14th International Conference on Hybrid Intelligent Systems*, pp. 23–29, 2014.
- [6] A. Helal B. Abdulrazak, “Tecarob : Tele-care using telepresence and robotic technology for assisting people with special needs,” *International Journal of ARM*, vol. 7, no. 3, pp. 46–53, 2006.
- [7] L. I. Oddsson, M. V. Radomski, M. White, D. Nilsson, “A robotic home telehealth platform system for treatment adherence, social assistance and companionship—an overview,” *2009 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 6437–6440, 2009.
- [8] M. Begum, R. Wang, R. Huq, A. Mihailidis, “Performance of daily activities by older adults with dementia : The role of an assistive robot,” *2013 IEEE 13th International Conference on Rehabilitation Robotics (ICORR)*, pp. 1–8, 2013.
- [9] T. S. Dahl M. N. K. Boulos, “Robots in health and social care : A complementary technology to home care and telehealthcare?,” *Robotics*, vol. 3, no. 1, pp. 1–21, 2013.
- [10] T.-C. Tsai, Y.-L. Hsu, A.-I. Ma, T. King, C.-H. Wu, “Developing a telepresence robot for interpersonal communication with the elderly in a home environment,” *Telemedicine and e-Health*, vol. 13, no. 4, pp. 407–424, 2007.
- [11] Y.-S. Chen, J.-M. Lu, Y.-L. Hsu, “Design and evaluation of a telepresence robot for interpersonal communication with older adults,” *ICOST 2013 : Inclusive Society : Health and Wellbeing in the Community, and Care at Home*, pp. 298–303, 2013.
- [12] A. Kristoffersson, S. Coradeschi, A. Loutfi, “A review of mobile robotic telepresence,” *Advances in Human-Computer Interaction*, vol. 2013, p. 3, 2013.
- [13] K. M. Tsui, M. Desai, H. Yanco, C. Uhlik, *et al.*, “Exploring use cases for telepresence robots,” *Proceedings ACM/IEEE International Conference Human-Robot Interaction*, pp. 11–18, 2011.
- [14] E. Guizzo, “When my avatar went to work,” *IEEE Spectrum*, vol. 9, no. 47, pp. 26–50, 2010.

- 
- [15] F. Michaud, F. Ferland, D. Létourneau, M.-A. Legault, M. Lauria, “Toward autonomous, compliant, omnidirectional humanoid robots for natural interaction in real-life settings,” *Paladyn, Journal of Behavioral Robotics*, vol. 1, no. 1, pp. 57–65, 2010.
  - [16] F. Ferland, *Conception d’un mécanisme intégré d’attention sélective dans une architecture comportementale pour robots autonomes*. PhD thesis, Département de génie électrique et de génie informatique, Université de Sherbrooke, 2014.
  - [17] F. Ferland, A. Reveleau, F. Leconte, D. Létourneau, F. Michaud, “Coordination mechanism for integrated design of human-robot interaction scenarios,” *Paladyn, Journal of Behavioral Robotics*, vol. 8, no. 1, 2017.
  - [18] M. Labbé F. Michaud, “Appearance-based loop closure detection for online large-scale and long-term operation,” *IEEE Transactions on Robotics*, vol. 29, pp. 734–745, June 2013.
  - [19] M. Labbé F. Michaud, “Long-term online multi-session graph-based SPLAM with memory management,” *Autonomous Robots*, pp. 1 – 18, 2017.
  - [20] F. Grondin F. Michaud, “Lightweight and optimized sound source localization and tracking methods for open and closed microphone array configurations,” *Robotics and Autonomous Systems*, vol. 113, pp. 63–80, 2019.
  - [21] “Double robotics - telepresence robot for telecommuters.” <http://www.doublerobotics.com/>. Consulté en février 2016.
  - [22] “About giraff.” <http://www.giraff.org/about-giraff/?lang=en>. Consulté en Août 2016.
  - [23] C. Major, “Video : Robots at work.” <http://www.nytimes.com/video/science/1248068965210/robots-at-work.html>. Consulté en février 2016.
  - [24] SuitableTechnologies, “Beam+ smart presence system.” <https://suitabletech.com/beam-plus/>. Consulté en février 2016.
  - [25] Anybots, “Qb telepresence robot.” <https://www.anybots.com/products/>. Consulté le 3 Mars 2016.
  - [26] Vgocom.com, “Establish your physical presence in a distant location with VGo | VGo robotic telepresence for healthcare, education and business.” <http://www.vgocom.com/what-vgo>. Consulté en février 2016.
  - [27] Crosswing, “Simulation of virtualme.” [http://photo.crosswing.com/shared/virtualME\\_Robot/virtualMe\\_Telepresence\\_Scenario\\_No\\_Intro\\_CrossWing.mp4](http://photo.crosswing.com/shared/virtualME_Robot/virtualMe_Telepresence_Scenario_No_Intro_CrossWing.mp4). Consulté en février 2016.
  - [28] “Gostai.” <http://www.gostai.com/>. Consulté en février 2016.
  - [29] “Prop.” <http://www.prop.org/>. Consulté en février 2016.
  - [30] “Mantarobot - teleme.” <http://www.mantarobot.com/products/teleme/features.htm>. Consulté en février 2016.
  - [31] “Personal robot group - mebot.” <http://robotic.media.mit.edu/portfolio/mebot/>. Consulté en février 2016.
  - [32] S. O. Adalgeirsson C. Breazeal, “Mebot : A robotic platform for socially embodied presence,” *5th ACM/IEEE International Conference on Human-Robot Interaction*, pp. 15–22, 2010.
-

- 
- [33] “irobot intouch remote presence robot.” <http://www.irobot.com/For-Business/RP-VITA.aspx>. Consulté en Février 2016.
- [34] “irobot ava 500 video collaboration robot.” <http://www.irobot.com/For-Business/Ava-500.aspx>. Consulté en Février 2016.
- [35] J. Gonzalez-Jimenez, C. Galindo, J. R. Ruiz-Sarmiento, “Technical improvements of the Giraff telepresence robot based on users’ evaluation,” *2012 IEEE RO-MAN : The 21st IEEE International Symposium on Robot and Human Interactive Communication*, pp. 827–832, 2012.
- [36] Q. Ma, Y. Zou, T. Zhang, “Study of service robot architecture based on middleware and abstract environment,” *2012 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pp. 1200–1205, 2012.
- [37] M. Göller, F. Steinhardt, T. Kerscher, J. M. Zöllner, R. Dillmann, “Proactive avoidance of moving obstacles for a service robot utilizing a behavior-based control,” *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5984–5989, 2010.
- [38] K. L. Luna-Gallegos, E. R. Palacios-Hernandez, S. Hernandez-Mendez, A. Marin-Hernandez, “A proposed software architecture for controlling a service robot,” *2015 IEEE International Autumn Meeting on Power, Electronics and Computing (ROPEC)*, pp. 1–6, 2015.
- [39] T. Sorell H. Draper, “Robot carers, ethics, and older people,” *Ethics and Information Technology*, vol. 16, no. 3, pp. 183–195, 2014.
- [40] N. Hu, Z. Lou, G. Englebiene, B. Kröse, *et al.*, “Learning to recognize human activities from soft labeled data,” *Proceedings Robotics : Science and Systems X*, 2014.
- [41] J. González-Jiménez, C. Galindo, C. Gutierrez-Castaneda, “Evaluation of a telepresence robot for the elderly : A spanish experience,” *IWINAC 2013 : Natural and Artificial Models in Computation and Biology*, pp. 141–150, 2013.
- [42] S. Coradeschi, A. Cesta, G. Cortellessa, L. Coraci, J. Gonzalez, L. Karlsson, F. Furfari, A. Loutfi, A. Orlandini, F. Palumbo, *et al.*, “Giraffplus : Combining social interaction and long term monitoring for promoting independent living,” *2013 6th International Conference on Human System Interactions (HSI)*, pp. 578–585, 2013.
- [43] R. Memmesheimer, V. Seib, G. Heuer, P. Schmidt, D. Thies, I. Mykhalchyshyna, J. Klöckner, M. Schmitz, N. Yann, N. G. Wettengel, *et al.*, “RoboCup 2016-homer@ UniKoblenz (Germany),” *Active Vision Group of the University of Koblenz-Landau*, 2015.
- [44] V. Seib, S. Manthe, R. Memmesheimer, F. Polster, D. Paulus, “Team homer unikoblenz approaches and contributions to the robocup home competition,” *RoboCup 2015 : RoboCup 2015 : Robot World Cup XIX*, pp. 83–94, 2015.
- [45] H. S. Vargas, E. Olmedo, A. D. Martinez, V. Poisot, A. Perroni, A. Rodriguez, O. Granillo, C. Merino, C. Peto, W. Antonio, “Project Donaxi HOME Service Robot,” *Applied Mechanics and Materials*, vol. 423, pp. 2817–2820, 2013.
- [46] F. Siepmann, L. Ziegler, M. Kortkamp, S. Wachsmuth, “Deploying a modeling framework for reusable robot behavior to enable informed strategies for domestic service robots,” *Robotics and Autonomous Systems*, vol. 62, no. 5, pp. 619–631, 2014.
-

- 
- [47] M. White, M. V. Radomski, M. Finkelstein, D. A. S. Nilsson, L. I. E. Oddsson, “Assistive/socially assistive robotic platform for therapy and recovery : patient perspectives,” *International Journal of Telemedicine and Applications*, vol. 2013, p. 11, 2013.
  - [48] C. Mucchiani, W. O. Torres, D. Edgar, M. J. Johnson, P. Z. Cacchione, M. Yim, “Development and deployment of a mobile manipulator for assisting and entertaining elders living in supportive apartment living facilities,” *2018 27th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, pp. 121–128, 2018.
  - [49] I. Mendez, M. Jong, D. Keays-White, G. Turner, “The use of remote presence for health care delivery in a northern Inuit community : A feasibility study,” *Int. J. Circumpolar Health*, vol. 72, 2013.
  - [50] E. Chien, M. Geffen, D. Gordon, “Telehealth benefits and adoption – connecting people and providers across canada,” *A Study Commissioned by Canada Health Infoway*, vol. 30, 2011.
  - [51] G. Pineau, K. Moqadem, C. St-Hilaire, E. Levac, B. Hamel, “Télésanté : Lignes directrices cliniques et normes technologiques en téléadaptation. agences des technologies et des modes d’intervention en santé,” tech. rep., Gouvernement du Québec, 2006.
  - [52] E. Ward, J. Crombie, M. Trickey, A. Hill, D. Theodoros, T. Russell, “Assessment of communication and swallowing post-laryngectomy : A telerehabilitation trial,” *Journal of Telemedicine and Telecare*, vol. 15, pp. 232–237, 2009.
  - [53] A. M. Joseph, “Care coordination and telehealth technology in promoting self-management among chronically ill patients,” *Telemedicine Journal E-Health : The Official Journal of American Telemedicine Association*, vol. 12, pp. 156–159, 2006.
  - [54] M. Tousignant, P. Boissy, H. Moffet, H. Corriveau, F. Cabana, F. Marquis, J. Simard, “Patients’ satisfaction of healthcare services and perception within-home telerehabilitation and physiotherapists’ satisfaction toward technology for post-knee arthroplasty : an embedded study in a randomized trial,” *J. Telemed. e-Health*, vol. 17, no. 5, pp. 376–382, 2011.
  - [55] T. Lewis, J. Drury, B. Beltz, “Evaluating mobile remote presence (MRP) robots,” *18th International Conference on Supporting Group Work*, pp. 302–305, 2014.
  - [56] H. A. Yanco, J. L. Drury, J. Scholtz, “Beyond usability evaluation : Analysis of human-robot interaction at a major robotics competition,” *Human-Computer Interaction*, vol. 19, no. 1-2, pp. 117–149, 2004.
  - [57] Y. Shaky M. J. Johnson, “A mobile robot therapist for under-supervised training with robot/computer assisted motivating systems,” *2008 30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 4511–4514, 2008.
  - [58] M. Takagi, Y. Takahashi, T. Komeda, “A universal mobile robot for assistive tasks,” *2009 IEEE International Conference on Rehabilitation Robotics*, pp. 524–528, 2009.
  - [59] T. Taipalus K. Kosuge, “Development of service robot for fetching objects in home environment,” *2005 International Symposium on Computational Intelligence in Robotics and Automation*, pp. 451–456, 2005.
  - [60] F. Grondin, D. Létourneau, F. Ferland, V. Rousseau, F. Michaud, “The Manyears open framework,” *Autonomous Robots*, vol. 34, no. 3, pp. 217–232, 2013.
-

- [61] F. Ferland F. Michaud, “Perceptual filtering for selective attention in a behaviour-based robot control architecture,” *IEEE Transactions on Cognitive and Developmental Systems*, vol. 8, no. 4, pp. 256 – 270, 2016.
  - [62] S. Laniel, D. Létourneau, M. Labbé, F. Grondin, J. Polgar, F. Michaud, “Adding navigation, artificial audition and vital sign monitoring capabilities to a telepresence mobile robot for remote home care applications,” *2017 International Conference on Rehabilitation Robotics (ICORR)*, pp. 809–811, July 2017.
  - [63] S. Coradeschi, A. Cesta, G. Cortellessa, L. Coraci, C. Galindo, J. Gonzalez, L. Karlsson, A. Forsberg, S. Frennert, F. Furfari, A. Loutfi, A. Orlandini, F. Palumbo, F. Pecora, S. V. Rump, A. Štítec, J. Ullberg, B. Ötslund, “Giraffplus : A system for monitoring activities and physiological parameters and promoting social interaction for elderly,” *Advanced Intelligent System Computing*, vol. 300, pp. 261–271, 2014.
  - [64] F. Palumbo, P. Barsocchi, F. Furfari, E. Ferro, “AAL middleware infrastructure for green bed activity monitoring,” *Journal of Sensors*, vol. 2013, 2013.
  - [65] L. Xiang, “Mapping and control with telepresence robots,” tech. rep., Department of Computer Science, Brown University, 2015.
  - [66] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, A. Y. Ng, “ROS : An open-source Robot Operating System,” in *ICRA Workshop on Open Source Software*, vol. 3, pp. 5–11, 2009.
  - [67] P. Lepage, D. Létourneau, M. Hamel, S. Brière, H. Corriveau, M. Tousignant, F. Michaud, “Telehomecare telecommunication framework – from remote patient monitoring to video visits and robot telepresence,” *2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 3269–3272, Aug 2016.
  - [68] C. Stachniss, *Robotic Mapping and Exploration*, vol. 55. Springer Science & Business Media, 2009.
  - [69] T. Moore D. Stouch, “A generalized extended Kalman filter implementation for the Robot Operating System,” *13th International Conference on Intelligent Autonomous Systems*, July 2014.
  - [70] G. Grisetti, S. Grzonka, C. Stachniss, P. Pfaff, W. Burgard, “Efficient estimation of accurate maximum likelihood maps in 3D,” *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3472–3478, Oct 2007.
  - [71] R. Wolff, M. Lasseck, M. Hild, O. Vilarroya, T. Hadzibeganovic, “Towards human-like production and binaural localization of speech sounds in humanoid robots,” *2009 3rd International Conference on Bioinformatics and Biomedical Engineering*, pp. 1–4, 2009.
  - [72] A. Reveleau, F. Ferland, F. Michaud, “Visual representation of interaction force and sound source in a teleoperation user interface for a mobile robot,” *Journal of Human-Robot Interaction*, vol. 4, no. 2, pp. 1–23, 2015.
  - [73] A. Magassouba, N. Bertin, F. Chaumette, “Audio-based robot control from interchannel level difference and absolute sound energy,” *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1992–1999, Oct 2016.
  - [74] C. Rascon I. Meza, “Localization of sound sources in robotics : A review,” *Robotics and Autonomous Systems*, vol. 96, pp. 184–210, 2017.
-

- 
- [75] V. Lunati, J. Manhès, P. Danès, “A versatile System-on-a-Programmable-Chip for array processing and binaural robot audition,” *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 998–1003, Oct 2012.
- [76] M. A. Goodrich A. C. Schultz, “Human-Robot Interaction : A survey,” *Foundation and Trends in Human-Computer Interaction*, vol. 1, pp. 203–275, <http://dx.doi.org/10.1561/11000000005>, Jan. 2007.
- [77] T. W. Fong, I. Nourbakhsh, K. Dautenhahn, “A survey of socially interactive robots,” *Robotics and Autonomous Systems*, vol. 42, pp. 143–166, 2003.
- [78] H. I. Christensen, T. Batzinger, K. Bekris, K. Bohringer, J. Bordogna, G. Bradski, O. Brock, J. Burnstein, T. Fuhlbrigge, R. Eastman, *et al.*, “A roadmap for us robotics : from internet to robotics,” *Computing Community Consortium*, vol. 44, 2009.
- [79] SPARC, “Robotics 2020 Multi-Annual Roadmap, for Robotics in Europe, Release B.” SPARC, The Partnership for Robotics in Europe (2015), Robotics 2020 Multi-Annual Roadmap, For Robotics in Europe, Release B, 2015.
- [80] G. N. Saridis, “Intelligent robotic control,” *IEEE Transactions on Automatic Control*, vol. AC-28, no. 5, pp. 547–557, <http://dx.doi.org/10.1109/TAC.1983.1103278>, 1983.
- [81] R. C. Arkin, *Behavior-Based Robotics*. MIT Press, 1998.
- [82] G. D. Konidaris G. M. Hayes, “An architecture for behavior-based reinforcement learning,” *Adaptive Behavior*, vol. 13, no. 1, pp. 5–32, 2005.
- [83] F. Ferland, A. Cruz-Maya, A. Tapus, “Adapting an hybrid behavior-based architecture with episodic memory to different humanoid robots,” *2015 24th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, pp. 797–802, 2015.
- [84] F. Ferland, R. Agrigoroaie, A. Tapus, “Assistive humanoid robots for the elderly with mild cognitive impairment,” *Humanoid Robotics : A Reference*, 2018.
- [85] Z. Kalal, K. Mikolajczyk, J. Matas, “Tracking-learning-detection,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 7, pp. 1409–1422, 2011.
-